MARCO TOMAMICHEL

# INFORMATION THEORY AND ITS APPLICATIONS

EE5139/EE6139 AT NUS

These notes are not yet free of typos and can always be improved. Any comments that help reduce these deficiencies are very much appreciated. I would like to thank all the students of NUS EE5139/EE6139 who have contributed to this effort already. Sections 0.2-0.3 and Section 8.2 are partially based on notes by Prof. Vincent Y. F. Tan. Most figures were contributed by Dr. Michael X. Cao. The exercises and solutions were developed together with Dr. Navneeth Ramakrishnan, Dr. Michael X. Cao, Yanglin Hu and Jan Seyfried.

http://www.marcotom.info/teaching/

*Latest update, May 2026*

# Contents

| | |
|---|---|
| $\emptyset$ | empty set $\{\}$ |
| $[M]$ | the set $\{1, 2, \ldots, M\}$ |
| $\lvert\mathcal{X}\rvert$ | cardinality of the set $\mathcal{X}$, e.g., $\lvert[M]\rvert = M$ |
| $\mathcal{P}(\mathcal{X})$ | the power set of $\mathcal{X}$, i.e. $\{A : A \subseteq \mathcal{X}\}$ |
| $\mathcal{X} \times \mathcal{Y}$ | set of tuples $\{(x, y) : x \in \mathcal{X}, y \in \mathcal{Y}\}$ |
| $\mathcal{X}^n$ | set of $n$-tuples with elements taking values in $\mathcal{X}$, |
| | e.g., $\{0,1\}^n$ is the set of $n$-bit strings |
| $\{0,1\}^*$ | the set of bit strings of arbitrary length |
| $x^n$ | the $n$-bit string $(x_1, x_2, \ldots, x_n)$ |
| $\max \mathcal{X}$ | largest $x^* \in \mathcal{X}$, might not always exist |
| $\sup \mathcal{X}$ | smallest $x^* \in \mathbb{R}$ such that $x \leq x^*$ for all $x \in \mathcal{X}$; |
| | equals the maximum, $\max \mathcal{X}$, if it exists |
| $\min \mathcal{X}$ | smallest $x^* \in \mathcal{X}$, might not always exist |
| $\inf \mathcal{X}$ | largest $x^* \in \mathbb{R}$ such that $x \geq x^*$ for all $x \in \mathcal{X}$; |
| | equals the minimum, $\min \mathcal{X}$, if it exists |
| $\mathbf{1}\{x = y\}$ | indicator function, 1 if $x = y$ and 0 otherwise, |
| | so that, for example, $\mathbf{1}\{x = y\} + \mathbf{1}\{x \neq y\} = 1$ |
| $\delta_{xy}$ | shorthand for $\mathbf{1}\{x = y\}$ |
| $P_X(x)$ | probability mass function (pmf), $P_X(x) = P[X = x]$ |
| $p_X(x)$ | probability density function (pdf), |
| | e.g., $P[X \in (1,2)] = \int_1^2 p_X(x)\mathrm{d}x$ |
| $P_X \times P_Y$ | the product distribution $P_{XY}(x, y) = P_X(x)P_Y(y)$ |
| $P_X^{\times n}$ | the i.i.d. distribution $P_{X^n}(x^n) = \prod_{i=1}^n P_X(x_i)$ |
| $P[X \in \mathcal{A}]$ | probability of $X$ being in a set $\mathcal{A}$, |
| | i.e., $P[X \in \mathcal{A}] = \mathbb{P}(\{\omega : X(\omega) \in \mathcal{A}\}) = \sum_{x \in \mathcal{A}} P_X(x)$ |
| $P[5 \leq X < 6]$ | another way of writing $P[X \in [5, 6)]$ |
| $\lceil x \rceil$ | ceiling operator, smallest $n \in \mathbb{N}$ such that $x \leq n$ |
| $\lfloor x \rfloor$ | floor operator, largest $n \in \mathbb{N}$ such that $x \geq n$ |
| log | logarithm; to base 2 here, i.e. $\log = \log_2$ |
| $\oplus$ | "XOR" function: $x \oplus y = 1 - \delta_{xy}$ for $x, y \in \{0, 1\}$; |
| | also addition in $F_2$, i.e. $x \oplus y = x + y \mod 2$. |

Table 1: Some basic notation used in these notes.

| | |
|---|---|
| pmf | probability mass function |
| pdf | probability density function |
| cdf | cumulative density function |
| rv | random variable |
| i.i.d. | independent and identically distributed |
| tvd | total variation distance |
| DMS | discrete memoryless source |
| DMC | discrete memoryless channel |
| DPI | data-processing inequality |
| LLR | log-likelihood ratio |
| AEP | asymptotic equipartition property |
| SNR | signal-to-noise ratio |
| iff | if and only if |

Table 2: Some abbreviations used in these notes.

# 0
# *Review of mathematical notation and foundations*

**Intended learning outcomes:**

- You are familiar with common notation used throughout the lecture.
- You are comfortable with the main mathematical concepts needed in this module, namely basic probability theory including random variables, conditional probabilities and Markov chains.
- You can apply basic bounds on tail probabilities.
- You understand the proof of the weak law of large numbers.
- You can compute vector norms and apply the Cauchy-Schwarz inequality.
- You know convex sets as well as convex and concave functions, and can apply Jensen's inequality.

## *0.1 Notation*

We will use standard notation and abbreviations that you might be familiar with from other modules. Some of the less frequently encountered mathematical expressions are summarised in Table 1 and the abbreviations we use throughout are listed in Table 2. (The tables can be found on the previous page.)

## *0.2 Probability theory*

We will not directly need the framework of probability theory in its most abstract formulation as presented in the following, but it is good to know that both discrete and continuous random variables can be seen as emanating from a shared mathematical framework.

### *0.2.1 Probability space*

A *probability space* is represented by a triple $(\Omega, \Sigma, \mathbb{P})$. Here $\Omega$ is a

set that is called the *sample space*. Moreover, $\Sigma$ is a $\sigma$-algebra, a collection of subsets of $\Omega$, called *events*, with the following properties:

- $\Omega \in \Sigma$
- If $A \in \Sigma$, then its *complement*, $A^c = \Omega \setminus A$ is also in $\Sigma$, i.e. $A^c \in \Sigma$.
- If $A_1, A_2, \ldots, A_n, \ldots \in \Sigma$, then $\bigcup_{i=1}^{\infty} A_i \in \Sigma$

Finally, the probability measure $\mathbb{P}$ is a function $\mathbb{P} : \Sigma \to [0,1]$ defined on the measurable space $(\Omega, \Sigma)$, and represents your "belief" about the events in $\Sigma$. In order for $\mathbb{P}$ to be called a probability measure, it must satisfy the following two properties:

1. $\mathbb{P}(\Omega) = 1$
2. For $A_1, A_2, \ldots \in \Sigma$ such that $A_i \cap A_j = \emptyset$ for all $i \neq j$, i.e. for mutually *disjoint* sets, we have

$$\mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(A_i) \tag{1}$$

Show that the above also implies that $\emptyset \in \Sigma$ and $\bigcap_{i=1}^{\infty} A_i \in \Sigma$.

Show that $0 \leq \mathbb{P}(A) \leq 1$ for every $A \in \Sigma$.

The tuple $(\Omega, \Sigma)$ is also called a measurable space. For example, set $\Omega = [0,1]$ and assume we are interested in the probability of subsets of $\Omega$ that are intervals of the form $[a,b]$ where $0 \leq a < b \leq 1$. Then we should also be able to say something about the probability of the union, intersections, complement and so on of such intervals. This is captured by the definition of a $\sigma$-algebra. Think of $\Sigma$ as the properties of $\Omega$ that can actually be observed.

**Example.** If your random variable is the location an athlete lands after a long jump then it makes sense to take $\Omega$ to be positive real numbers, $\mathbb{R}_+$ indicating the distance jumped (say, in meters). However, even with arbitrarily good equipment we cannot actually measure a real number, we can only ever say that he landed in some interval, the size of which is given by our measurement precision. Thus, $\Sigma$, comprised of the events we can actually observe, is built up by including all (arbitrarily small) intervals in $\mathbb{R}_+$ and their unions and complements. Note however that $\{x\} \in \Sigma$ for any $x \in \mathbb{R}^+$, that is, single points are also elements of the $\sigma$-algebra. Can you see why? Use an infinite intersection to construct it.

Some basic and very useful properties that can be derived from the above definition. The union bound in particular is very often used when analysing problems in information theory.

**Proposition 0.1** (Union Bound). *Let $(\Omega, \Sigma, \mathbb{P})$ be a probability space. The following holds:*

1. $\mathbb{P}(A^c) = 1 - \mathbb{P}(A)$
2. *If $A \subset B$, then $\mathbb{P}(A) \leq \mathbb{P}(B)$*
3. *We have the* union bound, $\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B)$. *Moreover, for any sets $A_1, A_2, \ldots \in \Sigma$, we have*

$$\mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) \leq \sum_{i=1}^{\infty} \mathbb{P}(A_i). \tag{2}$$

*Proof.* Property 1 follows since $A^c \cap A = \emptyset$, and thus $\mathbb{P}(A) + \mathbb{P}(A^c) = \mathbb{P}(\Omega) = 1$ by (1). For Property 2, note that $B \setminus A = B \cap A^c \in \Sigma$ and

since $A \cap (B \setminus A) = \emptyset$ we again argue that $\mathbb{P}(A) + \mathbb{P}(B \setminus A) = \mathbb{P}(B)$, from which the desired inequality follows.

For Property 3 note that $A \cup B$ can be decomposed in three different ways into mutually disjoint sets:

$$A \cup B = A \cup (B \setminus A) = B \cup (A \setminus B) = (A \setminus B) \cup (B \setminus A) \cup (A \cap B)\,. \tag{3}$$

Again using (1) for each of these decompositions we have

$$\begin{aligned} 2\mathbb{P}(A \cup B) &= \mathbb{P}(A) + \mathbb{P}(B) + \mathbb{P}(B \setminus A) + \mathbb{P}(A \setminus B) && (4)\\ &= \mathbb{P}(A) + \mathbb{P}(B) + \mathbb{P}(A \cup B) - \mathbb{P}(A \cap B)\,, && (5) \end{aligned}$$

which implies the desired equality. Clearly, by induction, the union bound works for finitely many sets $A_i, i = 1, \ldots, k$, namely

$$\mathbb{P}\left(\bigcup_{i=1}^{k} A_i\right) \leq \sum_{i=1}^{k} \mathbb{P}(A_i)\,. \tag{6}$$

and since $k$ can be arbitrarily large, this holds for any countable union of sets. □

Consider now two events $A, B \in \Sigma$. The conditional probability of $A$ given $B$ is defined as

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} \tag{7}$$

if $\mathbb{P}(B) \neq 0$. Two events $A, B \in \Sigma$ are said to be independent if

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B), \tag{8}$$

or, equivalently, if $\mathbb{P}(A|B) = \mathbb{P}(A)$ or $\mathbb{P}(B|A) = \mathbb{P}(B)$.

Verifty that $\mathbb{P}(\cdot|B)$ is indeed a probability measure on $(\Omega, \Sigma)$ according to the definition above.

Sometimes we have two conflicting beliefs, or models, about the underlying probability distribution, and so we will consider two compatible probability spaces $(\Omega, \Sigma, \mathbb{P})$ and $(\Omega, \Sigma, \mathbb{Q})$ based on the same measurable space $(\Omega, \Sigma)$. They offer different predictions about the probability with which the events in $\Sigma$ occur. A fundamental task in statistics is to deduce which model is correct from the frequency with which certain events occur. We will cover this in Chapter 3.

### 0.2.2 *Random variables*

We will usually not deal directly with the probability space but with random variables.

A *random variable* (rv) $X : \Omega \to \mathcal{X}$ is a function from the space $(\Omega, \Sigma)$ to a measurable space $(\mathcal{X}, \Sigma_X)$ satisfying $\{\omega \in \Omega : X(\omega) \in \mathcal{B}\} \in \Sigma$ for all $\mathcal{B} \in \Sigma_X$.

The mapping has to ensure that $\{\omega \in \Omega : X(\omega) \in \mathcal{B}\} \in \Sigma$ because we are restricted to observing events in $\Sigma$ and our random variable can thus not be more fine-grained than what $\Sigma$ allows. Functions satisfying this property are called *measurable functions*. A random variable is then more concisely defined as a measurable mapping from $(\Omega, \Sigma)$ to $(\mathcal{X}, \Sigma_X)$.

The probability measure $\mathbb{P}$ induces a probability measure $P_X$ on $(\mathcal{X}, \Sigma_X)$, given by

$$P_X(B) = \mathbb{P}(\{\omega \in \Omega : X(\omega) \in B\}) \tag{9}$$

for all $B \in \Sigma'$. $P$ is called the *distribution* of the random variable $X$. We often use a simplified notation that is more intuitive where we make the definition of the sets implicit; for example, $P[X \geq a]$ is a shorthand for $P_X(\{x \in \mathcal{X} : x \geq a\})$.

The only two examples of interest for us in the following are discrete and continuous random variables:

*discrete rv:* $\mathcal{X}$ is a discrete set and $\Sigma_X$ is the power set $\mathcal{P}(\mathcal{X})$ of $\mathcal{X}$, i.e. the set of all subsets of $\mathcal{X}$.

*continuous rv:* $\mathcal{X} = \mathbb{R}$ and $\Sigma_X = \mathcal{B}$, the Borel $\sigma$-algebra. This is the smallest $\sigma$-algebra containing all open intervals in $\mathbb{R}$.

If $\mathcal{X} = \{a_1, \ldots, a_d\}$ is discrete (and $\Sigma_X$ the power set of $\mathcal{X}$), then we say that $X$ is a *discrete random variable*. The distribution of $X$ is then also known as the *probability mass function (pmf)* of $X$ and is fully characterised by all the singleton events (consisting of a single value), i.e., the probabilities $P_X(a_1), P_X(a_2), \ldots, P_X(a_d)$.

Some discrete random variables are not random at all. If there is an $a_i$ with $P_X(a_i) = 1$ (and thus $P_X(a_j) = 0$ for all $j \neq i$), then we call this random variable *deterministic*. On the other extreme we have *uniformly distributed* random variables, where $P_X(a_i) = \frac{1}{d}$ for all $i \in [d]$.

Can you give a formal argument why the values at these points are sufficient?

**Example.** *The simplest example is the Bernoulli random variable. It is defined on a binary alphabet $\mathcal{X} = \{0, 1\}$ and we write $X \sim \text{Bern}(\epsilon)$ to denote the rv with $P[X = 1] = \epsilon$ and $P[X = 0] = 1 - \epsilon$.*

Let us now consider a real-valued random variable $X$. If there exists a function $p_X : \mathbb{R} \to [0, \infty)$ such that for all $A \in \Sigma_X$, we have

$$P[X \in A] = \int_A p_X(x)\, \mathrm{d}x \tag{10}$$

then we say that $X$ is a *continuous random variable*. The function $p_X$ is called the *probability density function (pdf)* of $X$. We also define the *cumulative distribution function* (cdf) by integrating $p_X(x)$, that is, the cdf is given by $F_X(a) = P[X \leq a] = \int_{-\infty}^{a} p_X(x)\,\mathrm{d}x$.

In this module, we deal mainly with discrete rvs, although we will also encounter Gaussian random variables, which are continuous, later on.

Show that $\int_{\mathcal{X}} p_X(x) = 1$. Moreover, if $p_X$ is continuous at some point $x$, it must satisfy $p_X(x) \geq 0$. Can $p_X(x)$ ever be larger than 1?

**Example.** *We denote the pdf of a* Gaussian *random variable $X$ as*

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{11}$$

*where $\mu$ is the* mean *and $\sigma$ the* standard deviation *of $X$. The* variance *of $X$ is $\sigma^2$. A* normal Gaussian *random variable has $\mu = 0$ and $\sigma = 1$ and its pdf is $\phi(x) = \mathcal{N}(x; 0, 1)$. The corresponding cdf is denoted as*

$$\Phi(y) = \int_{-\infty}^{y} \mathcal{N}(x; 0, 1)\,\mathrm{d}x\,. \tag{12}$$

### 0.2.3 Correlated random variables and independence

Here we focus on discrete random variables and use the respective notation. The counterparts for continuous random variables can be obtained by simply replacing pmfs with pdfs. Thus assume now that $X$ and $Y$ are discrete random variables taking on values in $\mathcal{X}$ and $\mathcal{Y}$ respectively.

- The joint pmf of $X$ and $Y$ is defined as

$$\begin{aligned} P_{X,Y}(x, y) &= P[X = x \wedge Y = y] \\ &= \mathbb{P}\big(\{\omega \in \Omega : X(\omega) = x \wedge Y(\omega) = y\}\big)\,. \end{aligned} \tag{13}$$

Verify that
$$P_Y(y) = \sum_{x' \in \mathcal{X}} P_{X,Y}(x', y).$$

- The conditional pmf is given by

$$P_{X|Y}(x|y) = \frac{P_{X,Y}(x, y)}{P_Y(y)}, \quad \text{for} \quad P_Y(y) > 0\,. \tag{14}$$

If $P_Y(y) = 0$ then the conditional pmf is ill-defined.

By applying the definition of conditional pmf twice, we arrive at Bayes' rule.

**Proposition 0.2** (Bayes' Rule). *Let $X$ and $Y$ be discrete and let $y \in \mathcal{Y}$ such that $P_Y(y) > 0$. Then,*

$$P_{X|Y}(x|y) = \frac{P_{Y|X}(y|x) P_X(x)}{P_Y(y)}\,. \tag{15}$$

Two random variables $X$ and $Y$ can be correlated, meaning that knowledge of $X$ will tell us something non-trivial about $Y$. We will make this more formal in an information-theoretic sense in the next chapter. Here, we only introduce the notion of independence, the absence of correlation.

Two random variables $X$ and $Y$ are *independent* if and only if, for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$,

$$P_{X,Y}(x,y) = P_X(x)P_Y(y) \tag{16}$$

or equivalently $P_{X|Y}(x|y) = P_X(x)$.

The latter condition simply states that the conditional distribution $P_{X|Y}(x|y)$ does not depend on $y$. In the case of two independent random variables $X$ and $Y$ we also say that their pmf is of *product* form, and we write $P_{XY} = P_X \times P_Y$.

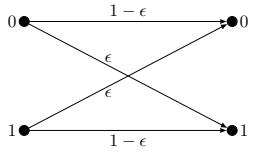
In the above we have defined the conditional pmf via the joint probability distribution. More often than not we will however go the other way: we will introduce a conditional probability distribution $W_{Y|X}$ and a marginal distribution $P_X$ to build up the joint distribution $P_{XY}(x,y) = P_X(x)W_{Y|X}(y|x)$. We call $W_{Y|X}$ a *channel* since it gives us a rule that tells us how to construct a random variable $Y$, the *channel output*, from a random variable $X$ with distribution $P_X$, the *channel input*. Channels are the basic object of study in information theory, and can be defined for discrete alphabets (as is done here) or continuous alphabets. We will cover them in more detail in Chapter 6.

**Example** (Binary symmetric channel)**.** *$X \sim \text{Bern}(p)$ is a bit that is sent over channel and is corrupted by additive noise $Z \sim \text{Bern}(\epsilon)$, where $X$ and $Z$ are independent. The output of the channel is $Y = X \oplus Z$. The channel is fully defined by the conditional distribution $P_{Y|X}$, which we can compute as*

$$\begin{aligned} P_{Y|X}(y|x) &= P[X \oplus Z = y \mid X = x] = P[Z = y \oplus x \mid X = x] \\ &= P[Z = y \oplus x] = P_Z(y \oplus x) \end{aligned} \tag{17}$$

*Hence, the channel can be given as a matrix or pictorially as follows:*

| $x$ | $y$ | $P_{Y\|X}$ |
|---|---|---|
| 0 | 0 | $1-\epsilon$ |
| 1 | 0 | $\epsilon$ |
| 0 | 1 | $\epsilon$ |
| 1 | 1 | $1-\epsilon$ |




What is the distribution of the channel output, $Y$, assuming that $X \sim \text{Bern}(p)$? Is it also Bernoulli, and if yes, with what parameter?

### 0.2.4 Expectation and variance

The expectation of a random variable $X$ is defined to be

$$\mathbb{E}[X] = \int_\Omega X(\omega)\,\mathrm{d}\mathbb{P}(\omega). \tag{18}$$

This definition has a very precise mathematical meaning in measure theory, but here we are only interested in two special cases. If $X$ is a discrete random variable this reduces to the familiar formula

$$\mathbb{E}[X] = \sum_{x\in\mathcal{X}} xP_X(x)\,. \tag{19}$$

If $X$ is a continuous random variable with pdf $p_X(x)$, we have

$$\mathbb{E}[X] = \int_{\mathbb{R}} xp_X(x)\,\mathrm{d}x\,. \tag{20}$$

Note that the expectation is a statistical summary of the distribution of $X$, rather than depending on the realised value of $X$. If there are two different models $\mathbb{P}$ and $\mathbb{Q}$ we need to specify which probability measure we are using. We only do this when necessary (because the model is not obvious from context) by adding a subscript $\mathbb{E}_P$ or $\mathbb{E}_Q$.

If $g$ is a function, the expectation of $g(X)$ is given by

$$\mathbb{E}[g(X)] = \int_{\mathbb{R}} g(x)p_X(x)\,\mathrm{d}x. \tag{21}$$

The variance of $X$ is the expectation of $g(X) = (X - \mathbb{E}[X])^2$. Thus,

$$\mathrm{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2] = \int_{\mathbb{R}} (x - \mathbb{E}[X])^2 p_X(x)\,\mathrm{d}x\,. \tag{22}$$

The conditional expectation $\mathbb{E}[X|Y]$ can be defined with respect to $P_{X|Y}$, but take note that this is actually a random variable as it depends on the value of $Y$ (a random variable). To get some intuition, suppose that $\mathcal{Y} = \{b_1, \ldots, b_m\}$. Then $\mathbb{E}[X|Y]$ is a random variable taking values $\mathbb{E}[X|Y = b_i]$, where $i = 1, \ldots, m$, with corresponding probability $P(Y = b_i)$. We see that $Y$ "partitions" the sample space $\Omega$ into different regions, and $\mathbb{E}[X|Y = b_i]$ is the normalised expectation of $X$ on the region corresponding to $Y = b_i$. We have

$$\mathbb{E}[X] = \sum_{i=1}^{m} \mathbb{E}[X|Y = b_i]P(Y = b_i) = \mathbb{E}[\mathbb{E}[X|Y]]. \tag{23}$$

In some sense, $\mathbb{E}[X|Y]$ is the "best" estimator of $X$ you can have, given knowledge of $Y$.

### 0.2.5 Markov chains

Markov chains describe a notion of conditional independence.

Show that the expectation is linear, i.e. $\mathbb{E}[aX + b] = a\mathbb{E}[X) + b$.

Check from the above definition that the variance can also be expressed as

$$\mathrm{Var}(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2.$$

Verify that $\mathcal{N}(x; \mu, \sigma^2)$ indeed has expectation $\mu$ and variance $\sigma^2$.

Assume $X - Y - Z$. Show that it is also true that $Z - Y - X$.

Let $X, Y$ and $Z$ be rvs. They are said to form a *Markov chain in the order* $X - Y - Z$ if their joint distribution $P_{XYZ}$ satisfies

$$P_{XYZ}(x,y,z) = P_X(x)P_{Y|X}(y|x)P_{Z|Y}(z|y) \tag{24}$$

for all $(x,y,z) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$. This the same as saying that $X$ and $Z$ are *conditionally independent given* $Y$.

Markov chains often make an appearance in communication settings. Consider for example a distribution $P_X$ and two channels $W_{Y|X}$ and $W_{Z|Y}$ that are applied in sequence. By the definition above it is obvious that $X - Y - Z$ form a Markov chain.

Notice that if we do not assume anything about the joint distribution $P_{XYZ}$, then it factorises as

If $Z$ is a deterministic function of $Y$, show that $X - Y - Z$ is true.

$$P_{XYZ}(x,y,z) = P_X(x)P_{Y|X}(y|x)P_{Z|XY}(z|x,y) \tag{25}$$

so what Markovianity in the order $X - Y - Z$ ensures us is that $P_{Z|XY}(z|x,y) = P_{Z|Y}(z|y)$, i.e., we can drop the conditioning on $X$. In essence all the information that we can learn about $Z$ is already contained in $Y$. No other information about $Z$ can be gleaned from knowing $X$ if we already know $Y$. Another way of saying this is that the conditional distribution of $X$ and $Z$ given $Y = y$ factorises as

$$P_{XZ|Y}(x,z|y) = P_{X|Y}(x|y)P_{Z|Y}(z|y). \tag{26}$$

Notice that this is in direct analogy to the situation where $X$ and $Z$ are (marginally) independent. Simply set $Y$ to be a deterministic random variable (with only one possible outcome) to recover the definition of independence.

If $X$ and $Z$ are conditionally independent given $Y$, this does not imply that $X$ and $Z$ are marginally independent (in general). Construct a counterexample.

## 0.3 *Tail and concentration bounds*

In this section, we summarise some bounds on probabilities that we use extensively in the sequel. More precisely, we are interested in showing that the probability of a random variable deviating too far from its expectation value is small, so-called tail bounds. Concentration bounds are even stronger and allow us to show that the probability mass is concentrated closely around the expectation value of certain random variables.

### 0.3.1 *Markov and Chebyshev bounds*

We start with the familiar Markov and Chebyshev inequalities.

**Proposition 0.3** (Markov's inequality)**.** *Let $X$ be a real-valued non-negative random variable with pdf $p_X$. Then for any $a > 0$, we have*

$$P[X \geq a] \leq \frac{\mathbb{E}[X]}{a}. \tag{27}$$

*Proof.* By the definition of the expectation, we have

$$\mathbb{E}[X] = \sum_{x \in \mathcal{X}} x P_X(x) \geq \sum_{x \in \mathcal{X}} \mathbf{1}\{x \geq a\} x P_X(x) \tag{28}$$

$$\geq a \sum_{x \in \mathcal{X}} \mathbf{1}\{x \geq a\} P_X(x) = a P[X \geq a]. \tag{29}$$

In which step is non-negativity of $X$ used? Can you do the proof also for discrete random variables?

and we are done.

□

Note that this bound only becomes nontrivial if $a$ exceeds the expectation value $\mathbb{E}[X]$.

If we let $X$ above be the non-negative random variable $(X - \mathbb{E}[X])^2$, we obtain Chebyshev's inequality.

**Proposition 0.4** (Chebyshev's inequality)**.** *Let $X$ be a real-valued random variable with mean $\mu$ and variance $\sigma^2$. Then for any $a > 0$, we have*

$$P\big[|X - \mu| \geq a\sigma\big] \leq \frac{1}{a^2}. \tag{30}$$

*Proof.* Let $X$ in Markov's inequality be the random variable $g(X) = (X - \mathbb{E}[X])^2$. This is clearly non-negative and the expectation of $g(X)$ is $\mathrm{Var}(X) = \sigma^2$. Thus, by Markov's inequality, we have

$$P\big[g(X) \geq a^2\sigma^2\big] \leq \frac{\sigma^2}{a^2\sigma^2} = \frac{1}{a^2}. \tag{31}$$

Now, $g(X) \geq a^2\sigma^2$ if and only if $|X - \mu| \geq a\sigma$ so the claim is proved.

□

Chebyshev's inequality is very general as it works for any random variable. If we have more structure we can make stronger statements.

Consider the standard Gaussian distribution. Is Chebyshev's inequality tight?

### *0.3.2 Concentration bounds for i.i.d. random variables*

We now consider a collection of real-valued random variables that are independent and identically distributed (i.i.d.). In particular, let $\boldsymbol{X} = (X_1, \ldots, X_k, \ldots)$ be a sequence of independent random variables where each individual $X_i$ has distribution $P$ with finite mean $\mu$ and finite variance $\sigma^2$.

**Proposition 0.5** (Weak Law of Large Numbers)**.** *Let $\boldsymbol{X}$ be an i.i.d. sequence as above. Then, for every $\epsilon > 0$, we have*

$$\lim_{n\to\infty} P\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| \geq \epsilon\right] = 0. \tag{32}$$

*I.e., the average $\frac{1}{n}\sum_{i=1}^{n} X_i$ converges to $\mu$ in probability.*

Note that for a sequence of random variables $\{S_n\}_{n=1}^{\infty}$, we say that this sequence *converges to a number $b \in \mathbb{R}$ in probability* if for all $\epsilon > 0$,

$$\lim_{n\to\infty} P\big[|S_n - b| \geq \epsilon\big] = 0. \tag{33}$$

We also write this as $S_n \xrightarrow{\mathrm{P}} b$. Contrast this to convergence of numbers: We say that a sequence of numbers $\{s_n\}_{n=1}^{\infty}$ *converges to a number $b \in \mathbb{R}$* if we have $\lim_{n\to\infty} |s_n - b| = 0$.

*Proof.* Note that it suffices to prove the result for the case where the $X_i$ have zero mean: if $X_i$ have mean $\mu$, we simply apply the result to the normalised random variable $X_i - \mu$.

Let $\frac{1}{n}\sum_{i=1}^{n} X_i$ take the role of $X$ in Chebyshev's inequality. Clearly, the mean is zero. The variance of $X$ is

$$\mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\mathrm{Var}\left(\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\sum_{i=1}^{n}\mathrm{Var}(X_i) = \frac{\sigma^2}{n}. \tag{34}$$

Thus, we have

$$P\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i\right| \geq \epsilon\right] \leq \frac{\sigma^2}{n\epsilon^2} \longrightarrow 0 \tag{35}$$

as $n \to \infty$, which proves the claim. □

In fact, we can say something even stronger, namely we can give tail bounds that vanish exponentially as we move further from the expectation value.

**Proposition 0.6** (Hoeffding's inequality)**.** *Let $\boldsymbol{X}$ as above, and furthermore $0 \leq X_i \leq 1$ for all $i \in \mathbb{N}$. Then, for all $\epsilon > 0$ and $n \in \mathbb{N}$,*

$$P\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| \geq \epsilon\right] \leq 2e^{-2n\epsilon^2}. \tag{36}$$

Some further tail bounds are discussed in Exercises 0.6 and 0.7. Another strengthening of the weak law of large numbers is the central limit theorem. If the scaling in front of the sum in the statement of the law of large numbers, Proposition 0.5, is $1/\sqrt{n}$ instead of $1/n$,

the resultant random variable $\frac{1}{\sqrt{n}} \sum_{i=1}^{n} X_i$ converges in distribution to a Gaussian random variable. As in Proposition 0.5, let $X^n$ be a collection of i.i.d. random variables where each $X_i$ is zero mean with finite variance $\sigma^2$.

**Proposition 0.7** (Central Limit Theorem)**.** *For any $a \in \mathbb{R}$, we have*

$$\lim_{n\to\infty} P\left[\frac{1}{\sigma\sqrt{n}} \sum_{i=1}^{n} X_i < a\right] = \Phi(a). \tag{37}$$

*In other words,*

$$\frac{1}{\sigma\sqrt{n}} \sum_{i=1}^{n} X_i \xrightarrow{\text{d}} Z \tag{38}$$

*where $\xrightarrow{\text{d}}$ means convergence in distribution and $Z$ is a standard Gaussian random variable.*

For a sequence of random variables $\{S_n\}_{n=1}^{\infty}$, we say that this sequence of random variables *converges in distribution* to another random variable $\bar{S}$ if

$$\lim_{n\to\infty} P[S_n < a] = P[\bar{S} < a] \tag{39}$$

for all $a \in \mathbb{R}$. The proof of this statement requires tools that are outside the scope of this course, but can be found in any textbook on probability theory.

## 0.4 Finite fields

A finite field is a field (on which addition, subtraction, multiplication and division are defined) with a finite number of elements. Such fields are denoted by $F_q$ where $q$ is the number of elements in the field, or its *dimension*. For each dimension, the field is unique up to a relabelling of the elements. The idea is that such fields behave like $\mathbb{Q}$, $\mathbb{R}$ or $\mathbb{C}$, with the usual rules for addition and multiplication, but are easier to manipulate on a digital computer.

Let us first formally define what a field is.

A *field* is a set $\mathbb{F}$ equipped with two binary operations denoted by $+$ and $\cdot$ that satisfy the following properties (here $a, b, c \in \mathbb{F}$ are arbitrary):

*Identities:* There exist two distinct elements $0, 1 \in \mathbb{F}$ such that $a + 0 = a$ and $a \cdot 1 = a$ for all $a \in F_q$.

*Additive inverse:* Every $a$ has an additive inverse in $\mathbb{F}$, denoted $-a$, such that $a + (-a) = 0$.

*Multiplicative inverse:* Every $a \neq 0$ has a multiplicative inverse in

$\mathbb{F}$, denoted $a^{-1}$, such that $a \cdot a^{-1} = 1$.

*Commutativity:* We have $a + b = b + a$ and $a \cdot b = b \cdot a$.

*Associativity:* We have $a + (b + c) = (a + b) + c$ and $a \cdot (b \cdot c) = (a \cdot b) \cdot c$.

*Distributivity:* We have $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$.

Finite fields exist only for particular numbers of elements, namely for prime powers. We will first construct these objects here (and argue that they satisfy the properties of a field for primes), and leave a formal proof to standard textbooks on the topic.

**Proposition 0.8.** *There exists a finite field $F_q$ with $|F_q| = q$ whenever $q = p^\ell$ for a prime $p$ and $\ell \in \mathbb{N}$.*

We will not give a proof of this statement, but nonetheless show how these fields can be constructed. For $F_q$ where $q$ is prime we can always simply denote the elements of $F_q$ by the integers $\{0, 1, \ldots, q-1\}$ and use integer addition and multiplication modulo $q$ as our operations. The properties above immediately follow, except for the existence of a multiplicative inverse, which requires a justification.[1]

Verify the above properties for $F_2$ and $F_3$. Can you find the inverse of 2 for a general $F_q$ with prime $q$? Use that $q + 1$ is even.

This strategy fails for $q = 4$, which is obviously not a prime. The problem is that using multiplication modulo 4 we get

$$2 \cdot 0 = 0 \mod 4, \qquad 2 \cdot 1 = 2 \mod 4, \tag{40}$$

$$2 \cdot 2 = 4 \mod 4 = 0, \qquad 2 \cdot 3 = 6 \mod 4 = 2, \tag{41}$$

and hence 2 does not have a multiplicative inverse.

When $q$ is a prime power $q = p^\ell$ we can derive the arithmetic instead using a polynomial ring. We give the construction here; but we do not attempt to show why it works or that it is unique. (This would require us to wade deeper into number theory.) First, we denote the elements of $F_q$ by strings of length $\ell$ with elements in $F_p$. In particular, if the underlying prime is 2, these are simply binary strings, e.g., $F_4 = \{00, 01, 10, 11\}$. We can then interpret these elements as polynomials of degree $\ell - 1$ with coefficients in $F_p$. For example, the polynomials corresponding to the four elements of $F_4$ given by $ij$ would be $i \cdot x + j$.

We can add these polynomials modulo $p$ for each coefficient individually, so in particular for $p = 2$, the base we most often encounter in our binary world, the negation of each element is just the element itself. For multiplication, we simply multiply the polynomials modulo an irreducible polynomial.[2] The choice of irreducible polynomial turns out not to matter — the resulting fields are equivalent up to relabelling of elements.

Verify that $x^2 + x + 1$ is indeed irreducible over $F_2$. Is it also irreducible over $F_3$?

[1] To verify the existence of the inverse, we only need to show that for every $a \neq 0$, we have $\{a \cdot b : b \in F_q\} = F_q$, and thus, there exists in particular one element $b$ such that $a \cdot b = 1$. To verify the aforementioned property, assume for the sake of contradiction that there exist $b > b'$ such that $a \cdot b - a \cdot b' = a \cdot (b - b') = 0 \mod p$. But the prime factors of $a$ and $b - b'$ do not contain $p$ since $a < p$ and $b - b' < p$. Therefore, we can never have $a \cdot b = a \cdot b' \mod p$ unless $b = b'$.

[2] An irreducible polynomial is one that cannot be decomposed into a product of lower-degree polynomials. In particular, it has no roots in $F_p$. This condition is also sufficient for polynomials of degree at most 3.

**Example.** *For $F_4$ we can take the irreducible polynomial to be $x^2 + x + 1$. So for the above labelings $\{00, 01, 10, 11\}$ of elements, we get, for example,*

$$10 \cdot 01 = 10 : \quad x \cdot 1 = x, \tag{42}$$

$$10 \cdot 10 = 11 : \quad x \cdot x = x^2 \mod x^2 + x + 1 = x + 1, \tag{43}$$

$$10 \cdot 11 = 01 : \quad x \cdot (x+1) = x^2 + x \mod x^2 + x + 1 = 1\,. \tag{44}$$

*Hence,* 10 *and* 11 *are multiplicative inverses of each other. The full addition and multiplication tables can then be written down as follows:*

| $+$ | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| 00 | 00 | 01 | 10 | 11 |
| 01 | 01 | 00 | 11 | 10 |
| 10 | 10 | 11 | 00 | 01 |
| 11 | 11 | 10 | 01 | 00 |

| $\cdot$ | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| 00 | 00 | 00 | 00 | 00 |
| 01 | 00 | 01 | 10 | 11 |
| 10 | 00 | 10 | 11 | 01 |
| 11 | 00 | 11 | 01 | 10 |

Similar constructions can be done for every prime power, and, quite importantly for practical applications, all of this arithmetic can be implemented efficiently on a digital computer, which is important once we consider finite fields to construct codes.

## *0.5 Vectors and norms*

Consider a $d$-dimensional vector space $\mathbb{F}^d$, where $\mathbb{F}$ is any field (e.g., $\mathbb{R}$). We can equip it with an *inner product*, denoted by $\langle \cdot, \cdot \rangle$. For two general vectors $u, v \in \mathbb{F}^d$, it evaluates to

$$\langle u, v \rangle = uv^T = \sum_{i=1}^{d} u_i v_i\,, \tag{45}$$

where $v^T$ denotes the transpose of the vector $v$, and is a column vector. It has an *orthonormal basis* $\{e_i\}_{i=1}^d$, with $\langle e_i, e_j \rangle = \delta_{ij}$.

We can naturally interpret pmf's on an alphabet with $d$ symbols as row vectors in a $d$-dimensional inner-product space. Without loss of generality we take the alphabet to be $\mathcal{X} = \{1, 2, \ldots, d\} = [d]$ and define the vector $p \in \mathbb{R}^d$ by its elements $p_x = P_X(x)$ for $x \in [d]$.

The following inequality for real vector spaces is extremely useful.

**Proposition 0.9** (Cauchy-Schwarz Inequality)**.** *Let $u, v \in \mathbb{R}^d$. Then,*

$$|\langle u, v \rangle|^2 \leq \langle u, u \rangle \langle v, v \rangle\,. \tag{46}$$

*with equality iff $u = \alpha v$ for some $\alpha \in \mathbb{R}$.*

*Proof.* We observe that

$$\langle u,u\rangle\langle v,v\rangle - |\langle u,v\rangle|^2 = \left(\sum_i u_i^2\right)\cdot\left(\sum_i v_i^2\right) - \left(\sum_i u_i v_i\right)^2 \tag{47}$$

$$= \frac{1}{2}\sum_{i,j} u_i^2 v_j^2 + u_j^2 v_i^2 - 2u_i v_i u_j v_j$$

$$= \frac{1}{2}\sum_{i,j}\left(u_i v_j - u_j v_i\right)^2 \geq 0\,. \tag{48}$$

Equality can only hold if $u_i v_j = u_j v_i$ for all pairs $(i,j)$, which only holds if $u_i = \alpha v_i$ for some constant $\alpha$. □

On these real vector spaces we can also define $p$-norms for $p \geq 1$ as

$$\|v\|_p = \left(\sum_{i=1}^{d} |v_i|^p\right)^{\frac{1}{p}} \tag{49}$$

We will encounter the 1-norm and the 2-norm, the latter being the usual Euclidian norm of the vector, i.e. $\|u\|_2 = \sqrt{\langle u,u\rangle}$. The following variant of the Cauchy-Schwarz inequality will be of use later.

**Corollary 0.10.** *Let $u, v \in \mathbb{R}^d$. Then,*

$$\|u\cdot v\|_1 \leq \|u\|_2\,\|v\|_2\,, \tag{50}$$

*where $\cdot$ is the element-wise product of the vectors, i.e. $(u\cdot v)_i = u_i v_i$.*

Using the above, show that

$$\|v\|_1 \leq \sqrt{d}\|v\|_2$$

for any $v \in \mathbb{R}^d$.

*Proof.* Define $k \in \mathbb{R}^d$ using $k_i = \overline{\operatorname{sgn}}(u_i v_i)$, where $\overline{\operatorname{sgn}}$ is the modified sign function, i.e.

$$\overline{\operatorname{sgn}}(x) = \begin{cases} -1 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}\,. \tag{51}$$

We then have $\langle k\cdot u, v\rangle = \sum_{x=1}^{d} |u_i v_i| = \|u\cdot v\|_1$. Moreover, since $\overline{\operatorname{sgn}}(x)^2 = 1$ for all $x \in \mathbb{R}$, the Cauchy-Schwarz inequality yields

$$\left|\langle k\cdot u, v\rangle\right| \leq \sqrt{\langle k\cdot u, k\cdot u\rangle\langle v,v\rangle} = \|u\|_2\|v\|_2\,, \tag{52}$$

concluding the proof. □

## *0.6 Convex sets and functions*

Convex sets and functions are the bread and butter of optimisation theory. It turns out that many information quantities are naturally cast as optimisation problems and thus some basic literacy in this area is useful.

### 0.6.1 *Convex and concave functions*

A function $f(x)$ is said to be *convex* on [a, b] if for all $x, y \in [a, b]$ and $\lambda \in [0, 1]$, we have

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y). \tag{53}$$

If no interval is given the function is meant to be convex on its full domain.

For example, the statement $\log(x)$ is concave should be understood as $\log(x)$ is concave on $(0, \infty)$.

The function $f$ is *strictly convex* if equality in (53) holds only if $\lambda = 0$ or 1, or $x = y$. The function $f$ is *concave* if $-f$ is convex, and *strictly concave* if $-f$ is strictly convex.

It is always useful to keep an image of a concave (e.g., log) and a convex (e.g., exp) in the back of your mind. It allows you to quickly determine in which direction Jensen's inequality (see below) goes, for example.

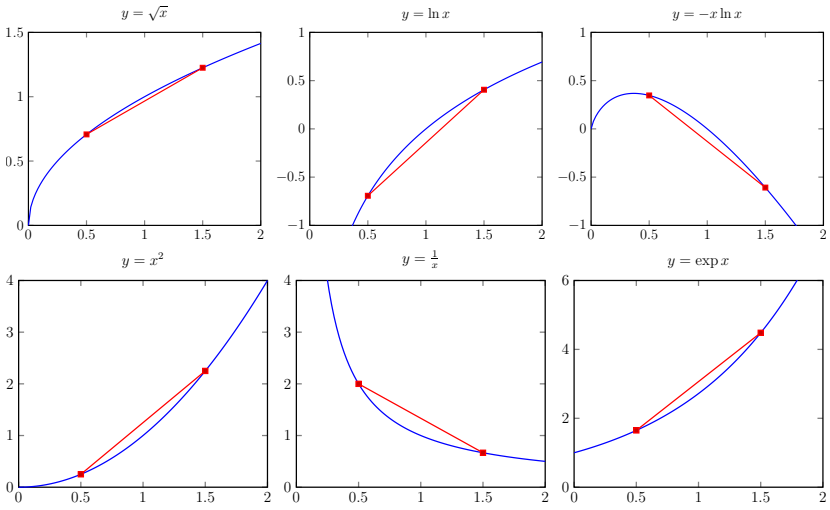



Figure 0.1: Examples of concave (upper line) and convex (lower line) functions. The straight line between two points of the curve is either below or above the plot of the function, which is exactly what the definition requires.

Often it is hard to check convexity directly. But for twice differentiable functions, there is a very direct way.

**Proposition 0.11.** *Let $f : [a, b] \to \mathbb{R}$ be twice differentiable. The function $f$ is convex if and only if $f''(x) \geq 0$ for all $x \in (a, b)$. Moreover, $f$ is strictly convex if $f''(x) > 0$ for all $x \in (a, b)$.*

*Proof.* We first prove the latter statement. Assume $f''(x) > 0$ for all $x \in [a, b]$. By Taylor expansion of $f$ around $x_0 \in (a, b)$, we have

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x^*)}{2}(x - x_0)^2 \tag{54}$$

Use this result to show that the function

$$f_1 : t \mapsto \log(t)$$

is concave and the function

$$f_2 : t \mapsto t \log(t)$$

is convex.

where $x^* \in [x_0, x]$. By assumption $f''(x^*) > 0$ so the quadratic term is strictly positive unless $x = x_0$, in which case it is still non-negative. Now let $x_0 = \lambda x_1 + (1-\lambda)x_2$. Further let $x = x_1$. Then we have

$$f(x_1) \geq f(x_0) + f'(x_0)((1-\lambda)(x_1 - x_2)). \tag{55}$$

Now let $x = x_2$. Then we have

$$f(x_2) \geq f(x_0) + f'(x_0)(\lambda(x_2 - x_1)). \tag{56}$$

Both of these inequalities are strict unless $\lambda \in \{0, 1\}$ or $x_1 = x_2$. Multiplying the first inequality by $\lambda$ and the second by $1-\lambda$ and adding them up, we recover the definition of strict convexity. If we instead had assumed only $f''(x) \geq 0$ the same argument would ensure convexity (but no longer strict convexity).

For the other direction, choose $a < x_1 < x_2 < x_3 < x_4 < b$. By the inequality to be shown in Exercise o.8, we have

$$\frac{f(x_2) - f(x_1)}{x_2 - x_1} \leq \frac{f(x_4) - f(x_3)}{x_4 - x_3} \tag{57}$$

Now let $x_2 \searrow x_1$ and $x_3 \nearrow x_4$. We see that $f'(x_1) \leq f'(x_4)$, and since these were arbitrary points, $f'$ is increasing on $(a, b)$. So $f''(x) \geq 0$ for all $x \in (a, b)$. □

### *o.6.2 Jensen's inequality*

Jensen's inequality is a direct result of our definition of convexity. It relates the value of a convex function $f$ at the point $\mathbb{E}[X]$ with the expectation value of $f(X)$.

**Proposition 0.12** (Jensen's inequality). *If $f(x)$ is convex and $X$ is a random variable on $\mathbb{R}$, then*

$$f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)] \tag{58}$$

We only give a proof for discrete distributions here.

*Proof.* We give a proof by induction. Due to convexity, we have

$$p_1 f(x_1) + p_2 f(x_2) \geq f(p_1 x_1 + p_2 x_2), \tag{59}$$

which proves the statement if $|\mathcal{X}| = 2$.

Suppose the statement $\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$ is true when $|\mathcal{X}| = k - 1$. Then consider a pmf with $k$ mass points $\{p_1, p_2, \ldots, p_k\}$. Define another pmf on $k-1$ points given by the probabilities

$$p'_i = \frac{p_i}{1 - p_k}, \quad i = 1, \ldots, k-1\,. \tag{60}$$

We then have

$$\sum_{i=1}^{k} p_i f(x_i) = p_k f(x_k) + (1-p_k) \sum_{i=1}^{k-1} p'_i f(x_i) \tag{61}$$

$$\geq p_k f(x_k) + (1-p_k) f\left(\sum_{i=1}^{k-1} p'_i x_i\right) \tag{62}$$

$$\geq f\left(p_k x_k + (1-p_k) \sum_{i=1}^{k-1} p'_i x_i\right) \tag{63}$$

$$= f\left(\sum_{i=1}^{k} p_i x_i\right) \tag{64}$$

where the first inequality is from the induction hypothesis and the second by convexity (of two points). By the definition of expectation we have $\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$. □

### *0.6.3 Convex sets and a minimax theorem*

Consider a set $\mathcal{X}$ that is defined on some linear space, for example a vector space.

A set $\mathcal{X}$ is *convex* if, for any $x_1, x_2 \in \mathcal{X}$, we have

$$\lambda x_1 + (1-\lambda) x_2 \in \mathcal{X}, \quad \forall \lambda \in (0,1)\,. \tag{65}$$

The set $\mathcal{P}(\mathcal{X})$ of probability distributions over some set $\mathcal{X}$ is convex, for example.

Verify that $\mathcal{P}(\mathcal{X})$ is convex.

With many optimisation problems it is often extremely useful to know whether the objective function (the function we minimise or maximise) is convex or concave in the variable since this implies that more or less efficient numerical solvers exist, at least when the sets we optimise over are also convex. Moreover, we cannot get stuck in a local optimum for such functions. Strict convexity or concavity even gives us a guarantee that solutions are unique.

Beyond that, the following theorem turns out to be useful for our deliberations in later chapters. We state it in a simplified and refer to the Exercise 0.9 for an example, but note that it is a special case of Sion's minimax theorem[3]. It is also worth noting that this is closely related to the notion of a Nash equilibrium in game theory, where a nash equilibrium is exactly a saddle point for the minimax function.

[3] Maurice Sion. On General Minimax Theorems. *Pacific Journal of Mathematics*, 8:171–176, 1958

**Proposition 0.13** (Minimax)**.** *Consider two compact convex sets $\mathcal{X}$ and $\mathcal{Y}$ and a continuous function $f : (\mathcal{X}, \mathcal{Y}) \to \mathbb{R}$ that is convex in the*

*first argument and concave in the second argument. Then,*

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y) = \max_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} f(x,y), \tag{66}$$

*i.e., the minimum and maximum can be interchanged.*

For a function $f(x,y)$ to be convex in the first argument, we require that $x \mapsto f(x,y)$ is convex on $\mathcal{X}$ for each $y \in \mathcal{Y}$. Note that the inequality $\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y) \geq \max_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} f(x,y)$ always holds independently of the objective function and the sets we optimise over. This is because the inner optimisation dominates the outer optmisation in the sense that the inner optimiser can be chosen as a function of the variable in the outer optimisation.

An example of a function that satisfies the above conditions is $f(x,y) = x^2(1-y^2)$ on $[-1,1]^{\times 2}$.

## 0.7 Exercises

**Exercise 0.1.** *Let $V$ and $W$ be discrete random variables defined on some probability space with a joint pmf $P_{VW}(v,w)$. We do not assume independence.*

a) *Prove that $\mathbb{E}[V+W] = \mathbb{E}[V] + \mathbb{E}[W]$.*

b) *Prove that if $V$ and $W$ are independent, then $\mathbb{E}[VW] = \mathbb{E}[V]\mathbb{E}[W]$.*

c) *Let $V$ and $W$ be independent and let $\sigma_V^2$ and $\sigma_W^2$ be their respective variances. Find the variance of $Z = V + W$.*

**Exercise 0.2.** *For a nonnegative integer-valued random variable $N$, show that $\mathbb{E}[N] = \sum_{n>0} P[N \geq n]$ and that $P[N \geq n] = \mathbb{E}[\mathbf{1}\{N \geq n\}]$.*

**Exercise 0.3.** *Flip a fair coin four times. Let $X$ be the number of Heads obtained, and let $Y$ be the position of the first Heads i.e. if the sequence of coin flips is TTHT, then $Y = 3$, if it is THHH, then $Y = 2$. If there are no heads in the four tosses, then we define $Y = 0$.*

a) *Model the experiment completely, i.e. define the sample space and the random variables $X$ and $Y$ as functions from that sample space.*

b) *Find the joint pmf of $X$ and $Y$.*

c) *Using the joint pmf, find the marginal pmf of $X$. What is $P[Y = 0|X = 1]$ and $P[Y = 1|X = 3]$?*

**Exercise 0.4.** *Derive the addition and multiplication tables for $F_8$ and $F_9$. You should use the construction described in the lecture notes and the irreducible polynomials $x^3 + x + 1$ for $F_8$ and $x^2 + 1$ for $F_9$.*

**Hint:** You may want to use MATLAB to solve this problem. However, you will need to compute some elements by hand to verify the computer-generated output.

**Exercise 0.5.** *For discrete random variables $A$ and $B$, derive the Cauchy-Schwarz inequality in the form*

$$\mathbb{E}[AB] \leq \sqrt{\mathbb{E}[A^2]\mathbb{E}[B^2]}$$

*from its vector form in Proposition 0.9.*

**Exercise 0.6.** *In this exercise we will derive some further tail bounds.*

a) *Derive the one-sided Cheybyshev inequality, which says that $P[Y \geq a] \leq \sigma_Y^2/(\sigma_Y^2 + a^2)$ if $\mathbb{E}[Y] = 0$ and $a > 0$.*

b) *Derive the reverse Markov inequality: Let $X$ be a random variable such that $P[X \leq a] = 1$ for some constant $a$. Then for $d < \mathbb{E}[X]$, we have*

$$P[X > d] \geq \frac{\mathbb{E}[X] - d}{a - d}.$$

**Exercise 0.7** (Chernoff bound)**.** *Let $X_1, \ldots, X_n$ be a sequence of i.i.d. rvs with zero-mean and moment generating function $M_X(s) := \mathbb{E}[e^{sX}]$. Show that for any $\epsilon > 0$,*

$$P\left[\frac{1}{n}(X_1 + \ldots + X_n) > \epsilon\right] \leq \exp\Big(-n \max_{s \geq 0}(\epsilon s - \log M_X(s))\Big).$$

**Hint:** Note that the event $\{\frac{1}{n}(X_1 + \ldots + X_n) > \epsilon\}$ occurs exactly if $\{\exp(s(X_1 + \ldots + X_n)) > \exp(ns\epsilon)\}$, for any fixed $s \geq 0$. Now apply Markov's inequality.

**Exercise 0.8.** *Let $f$ be convex on $[a, b]$. Using only the defining property of convex functions (see Section 0.6), show that for any $a \leq x_1 < x_2 \leq x_3 < x_4 \leq b$, we have*

$$\frac{f(x_2) - f(x_1)}{x_2 - x_1} \leq \frac{f(x_4) - f(x_3)}{x_4 - x_3}.$$

**Exercise 0.9** (Minimax Theorem)**.** *In this exercise, we aim to get an intuition for the Minimax Theorem in (66). We fix the function to be*

$$f(x, y) = x^2 + 4xy - y^2.$$

a) *Show that $f$ satisfies the requirements of the theorem and compute $g_{\mathcal{Y}}(x) := \max_{y \in \mathcal{Y}} f(x, y)$ and $h_{\mathcal{X}}(y) := \min_{x \in \mathcal{X}} f(x, y)$ for general intervals $\mathcal{X}$ and $\mathcal{Y}$, that is $\mathcal{X} = [x_{\min}, x_{\max}]$ and $\mathcal{Y} = [y_{\min}, y_{\max}]$ for variables $x_{\min} \leq x_{\max}$ and $y_{\min} \leq y_{\max}$. This requires a case distinction.*

b) *Convince yourself that the theorem holds for the specific choice $\mathcal{X} = [-2, 2]$ and $\mathcal{Y} = [-1, 1]$ by computing both sides of the equality starting from the result in a). Note that one side is more direct than the other.*

c) *Next, we modify $\mathcal{X}$ such that the result of the optimization changes. Let $\mathcal{X} = [x_{\min}, 2]$, where $2 > x_{\min} > 0$, and $\mathcal{Y} = [-1, 1]$. Calculate both sides of (66) as a function of $x_{\min}$.*

# 1
# *Information measures*

**Intended learning outcomes:**

- You can compute the entropy and conditional entropy for any discrete random variable and understand the basic properties of these two quantities, e.g. you can apply the chain rule or sub-additivity.
- You can compute mutual information and now how it relates to entropy and conditional entropy. You can apply the data-processing inequality for mutual information.
- You can compute the relative entropy and understand how entropy and mutual information can be expressed in terms of the relative entropy.

**Book reference:** Chapter 2 in Cover & Thomas[1], but we are not following it too closely.

[1] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991. ISBN 9780471748823. DOI: 10.1002/047174882X

## *1.1 Surprisal and entropy*

It is not immediately clear how to model our intuitive notion of "information" in a mathematical language. In this chapter we take a somewhat axiomatic approach to information measures, i.e. we try to build them up from our intuitive understanding of what entropy and information should be. But we will only really be able to justify the choices we make here once we start analysing practical problems in information theory, and see that the quantities we derive and investigate here pop up again and again as solutions.

### *1.1.1 Surprisal*

It turns out to be fruitful to start not by finding an expression for the information contained in a random variable, but rather the lack of information, or uncertainty inherent in a random experiment. Let us consider a discrete random variable $X$ taking values in $\mathcal{X}$ following the pmf $P_X(x) = p_x$. How surprised are we to see a particular outcome $x \in \mathcal{X}$ of this random experiment? Clearly this depends

on the probability $p_x$ and not the value of $x$ itself. In fact, we do not even need to know what $\mathcal{X}$ really is. On the one hand, if $p_x = 1$ we are not surprised at all since we already knew that we would see $x$. On the other hand, the smaller $p_x$ is the more surprised we are to see this particular outcome. If $p_x = 0$ we will never see $x$, so our surprise when seeing it anyway would be literally off the scale. Furthermore — and this turns out to be a very convenient choice — if we do a random experiment twice independently and both times observe $x$, we say that we will be twice as surprised as if we had seen $x$ once in a single random experiment.

The above notions can be formalised, and that is essentially what Shannon did when he introduce the notion of *surprisal*.[2] Let us denote the surprisal of $x$ as $s(p_x)$. We want this function to satisfy the following three conditions:

[2] C. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 1948. DOI: 10.1002/j.1538-7305.1948.tb00917.x

1. **Monotonicity:** $s(p_x) = 0$ if $p_x = 1$ and $s(p_x)$ increases monotonically as $p_x$ decreases.
2. **Additivity:** The surprisal of seeing a pair of outcomes of independent random experiments is simply the sum of the individual surprisals, i.e. $s(p_x p_y) = s(p_x) + s(p_y)$.
3. **Normalisation:** Each outcome of a fair coin toss has unit surprisal, i.e., $s(\frac{1}{2}) = 1$.

We do not really need the condition $s(p_x) = 0$ if $p_x = 1$ under Point 1 as it follows directly from additivity. Can you see how?

It turns out that the only positive function that satisfies the first two properties is the logarithm. To show this one uses a result by Erdös that characterises additive functions, but that is beyond the scope here. We therefore pick

$$s(p_x) = \log \frac{1}{p_x}\,. \tag{1.1}$$

where the logarithm is taken to base 2 (as everywhere in these notes) so that the normalisation requirement is also satisfied.

We can see the surprisal as another random variable, say $S$, that takes the value $s(p_x) = \log \frac{1}{p_x}$ with probability $p_x$. Since $S = S(X)$ is a function of $X$ we usually simply write this new random variable as

$$S(X) = \log \frac{1}{P_X(X)}\,. \tag{1.2}$$

### 1.1.2 *Entropy*

Entropy measures how much we can learn by looking at the outcome of a random experiments, or, in other words, how much uncertainty there is about the outcome. It is simply the expected surprisal of $X$.

Given a discrete random variable $X$, the *entropy* of $X$ is defined as

$$H(X) := \mathbb{E}[S(X)] = \mathbb{E}\left[\log \frac{1}{P_X(X)}\right] = \sum_{x \in \mathcal{X}} p_x \log \frac{1}{p_x} \tag{1.3}$$

Here and throughout we use the convention that $0 \log 0 = 0$. This is reasonable since $\lim_{\epsilon \to 0} \epsilon \log \epsilon = 0$, and thus we simply continuously extend the function to the point 0.

Verify that $\epsilon \log \epsilon \to 0$ as $\epsilon \to 0$.

Note again that the entropy $X$ is really only a function of the pmf of $X$, and in particular independent of the alphabet $\mathcal{X}$, in contrast to potential alternative uncertainty measures like the variance of $X$. In some cases we want to emphasise the underlying pmf, let us say $P_X$ or $Q_X$, and will thus write $H(X)_P$ and $H(X)_Q$, but in most cases we can drop this subscript because the distribution is clear from context.

Sometimes we are interested in more than just the expected surprisal. The minimum surprisal, or min-entropy, for example, has applications in cryptography (see Chapter 4) and the variance of $S(X)$ has itself operational meaning in many information-theoretic problems when we go beyond first order asymptotics.

Can you find an expression for $\mathrm{Var}[S(X)]$ in terms of the probabilities $p_x$?

Now let us explore the entropy a bit. First we want to show the following basic property.

**Proposition 1.1.** *Let $X$ be a discrete random variable taking values in $\mathcal{X}$. We have*

$$H(X) \geq 0, \tag{1.4}$$

*with equality iff $X$ is deterministic.*

*Proof.* Since $p_x \leq 1$, we have $\log \frac{1}{p_x} \geq 0$ for every $x \in \mathcal{X}$, so the expectation of this quantity over $x$ must be non-negative too. In fact, $\log \frac{1}{p_x}$ equals 0 iff $p_x = 1$ and hence $H(X) = 0$ only if there exists an $x \in \mathcal{X}$ for which $p_x = 1$, which is the hallmark of a deterministic rv. □

The entropy is a strictly concave function of the pmf $P_X$. To see this, we first verify that $f(t) = t \log \frac{1}{t} = -t \log t$ is concave on $(0, 1)$ by taking its second derivative:

$$f'(t) = -\log t - \log e, \qquad f''(t) = -\frac{\log e}{t}. \tag{1.5}$$

Since the latter is always negative for $t \in (0, 1)$, the function is indeed strictly concave according to Lemma 0.11. Now since the entropy is simply the sum $\sum_{x \in \mathcal{X}} f(p_x)$ it is indeed a strictly concave function of the pmf. This simple property, together with Jensen's inequality, has profound implications. The first one is that the entropy has a unique

maximum. Intuitively we would want that entropy is maximal when uncertainty about the outcome is greatest, namely when the rv is uniformly distributed. And this is indeed the case.

**Proposition 1.2.** *Let $X$ be a discrete random variable taking values in $\mathcal{X}$. We have*

$$H(X) \leq \log |\mathcal{X}|, \tag{1.6}$$

*with equality iff $X$ is uniformly distributed.*

The general case will be covered in Exercise 1.1 but here we give a proof for the binary case when the set $\mathcal{X}$ is a bit.

Show that $h(p) = h(1-p)$.

*Proof for $\mathcal{X} = \{0,1\}$.* It is easy to verify by a simple computation that $H(X) = 1$ for a uniformly distributed random variable, so the difficulty is only in showing that this is the maximum and only achieved for the uniform distribution.

Let now $\{p, 1-p\}$ for $p \in [0,1]$ be a general pmf for the binary random variable $X$. We use the function $f(t) = -t \log t$ to simplify notation. Then we can write

$$\begin{aligned} H(X) &= f(p) + f(1-p) && (1.7)\\ &= \frac{1}{2}\left(f(p) + f(1-p)\right) + \frac{1}{2}\left(f(p) + f(1-p)\right) && (1.8)\\ &\leq f\left(\frac{1}{2}p + \frac{1}{2}(1-p)\right) + f\left(\frac{1}{2}p + \frac{1}{2}(1-p)\right) && (1.9)\\ &= f\left(\frac{1}{2}\right) + f\left(\frac{1}{2}\right) && (1.10)\\ &= 1\,. && (1.11) \end{aligned}$$

The inequality is due to Jensen's inequality. Moreover, due to the strict concavity of $f$ equality holds only if $p = 1-p = \frac{1}{2}$, i.e., when the random variable $X$ follows the uniform distribution. □

Concavity in fact has even stronger consequences, and we will show a few additional properties of entropy later on using it.

**Example.** *The simplest example of a random variable is the Bernoulli random variable $X \sim \mathrm{Bern}(p)$ with $\mathcal{X} = \{0,1\}$ and $P_X(0) = p$ for $p \in [0,1]$. The entropy of the Bernoulli random variable is called the* binary entropy*,*

$$H(X) = p \log \frac{1}{p} + (1-p) \log \frac{1}{1-p} =: h(p)\,. \tag{1.12}$$

*From the plot we can easily verify all the properties we discussed above.*

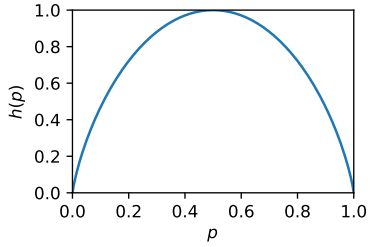



Figure 1.1: The binary entropy function.

## 1.2 Joint and conditional entropy

### 1.2.1 Joint entropy

For two discrete random variables $X$ and $Y$ with joint pmf $P_{XY}(x,y) = p_{xy}$ we can simply consider $(X,Y)$ as one single random variable and use the same construction to define the surprisal of a tuple $(X,Y)$ as $S(X,Y) = -\log P_{XY}(X,Y)$. Its expectation is the *joint entropy*, $H(XY)$, given by

$$H(XY) := \mathbb{E}[S(X,Y)] = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log\frac{1}{p_{xy}} \tag{1.13}$$

Find an example for which $H(XY) = H(X) = H(Y) = 1$.

The first thing to note is that — if $X$ and $Y$ are independent — then $p_{xy} = p_x \cdot p_y$ and thus the expression simplifies to

$$\begin{aligned} H(XY) &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log\frac{1}{p_x p_y} && (1.14)\\ &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\left(\log\frac{1}{p_x}+\log\frac{1}{p_y}\right) && (1.15)\\ &= \sum_{x\in\mathcal{X}} p_x\log\frac{1}{p_x} + \sum_{y\in\mathcal{Y}} p_y\log\frac{1}{p_y} && (1.16)\\ &= H(X)+H(Y)\,. && (1.17)\end{aligned}$$

This is not true in general though if the two random variables are correlated.

### 1.2.2 Conditional entropy

So why do these entropies not just add up in the presence of correlations? Fundamentally, this is because once we learn $X$ we might not be so surprised seeing some particular outcomes of the random variable $Y$ anymore. In fact, in the most extreme case, we have $Y = f(X)$ for some function $f$; hence, once we know that $X$ takes on the value $x$, we can immediately deduce that $Y$ will take on the value $f(x)$ with probability one, and thus there is no surprisal anymore! We model this "conditional suprisal" using the conditional pmfs, $P_{Y|X}(y|x) = p_{y|x}$, which leads us to conditional entropy.

The *conditional entropy* of $Y$ given $X$ is defined as

$$H(Y|X) = \mathbb{E}\left[\log\frac{1}{P_{Y|X}(Y|X)}\right] = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log\frac{1}{p_{y|x}}\,. \tag{1.18}$$

This can be interpreted as the expectation of the entropy of $Y$ over all outcomes $X$. We sometimes use the notation $H(Y|X=x)$ to denote

the entropy $\{p_{y|x}\}_{y\in\mathcal{Y}}$, i.e., the pmf of $Y$ when we already know that $X = x$. Using this and the expression in (1.18) we can write the conditional entropy as

$$H(Y|X) = \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{xy} \log \frac{1}{p_{y|x}} \tag{1.19}$$

$$= \sum_{x\in\mathcal{X}} p_x \sum_{y\in\mathcal{Y}} p_{y|x} \log \frac{1}{p_{y|x}} \tag{1.20}$$

$$= \sum_{x} p_x \, H(Y|X = x) \,. \tag{1.21}$$

The last line which expresses the conditional entropy in terms of an average of (unconditional) entropies is particularly useful since it allows us to immediately conclude that the conditional entropy is also bounded from below and above, like the entropy. We thus have

$$0 \leq H(Y|X) \leq \log |\mathcal{Y}| \,. \tag{1.22}$$

Moreover, our definition of conditional entropy also allows us to establish a *chain rule* for the conditional entropy, which sometimes is in fact used as the definition of conditional entropy itself. This rule is very useful because it allows us to write the joint entropy as a sum of its parts, even if the two random variables are not independent.

**Proposition 1.3** (Chain Rule). *We have* $H(XY) = H(X) + H(Y|X)$.

*Proof.* We take advantage of $p_{xy} = p_x p_{y|x}$ to write

$$H(XY) = \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{xy} \log \frac{1}{p_{xy}} \tag{1.23}$$

$$= \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{xy} \log \frac{1}{p_x} + \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{xy} \log \frac{1}{p_{y|x}} \tag{1.24}$$

$$= \sum_{x\in\mathcal{X}} p_x \log \frac{1}{p_x} + \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{xy} \log \frac{1}{p_{y|x}} \tag{1.25}$$

$$= H(X) + H(Y|X) \,. \tag{1.26}$$

□

Show that $H(Y|X) = H(Y)$ for independent random variables. Using the chain rule, find a different proof that $H(XY) = H(X) + H(Y)$ in this case.

### 1.2.3 *Sub-additivity*

Now we have put everything in place to show our first entropic inequality, which relates the entropy of two random variables with their joint entropy. This result shows the *sub-additivity* of entropy.

**Proposition 1.4** (Sub-Additivity). *Let $X$ and $Y$ be two discrete random variables. Then*

$$H(XY) \leq H(X) + H(Y)\,, \tag{1.27}$$

*or, equivalently, $H(X|Y) \leq H(X)$. Equality holds in either statement iff $X$ and $Y$ are independent.*

*Proof.* The equivalence of the two relations follows directly from the chain rule, we thus only need to show the second statement. We already know from Eq. (1.17) that equality hold if $X$ and $Y$ are independent. It remains to show that the inequality holds, and that it holds with equality only if $X$ and $Y$ are independent.

We start with Eq. (1.21), which states that

$$\begin{aligned} H(Y|X) &= \sum_x p_x H(Y|X=x) && (1.28)\\ &= \sum_x p_x \sum_y p_{y|x} \log \frac{1}{p_{y|x}} && (1.29)\\ &= \mathbb{E}\left[\sum_y p_{y|X} \log \frac{1}{p_{y|X}}\right] && (1.30) \end{aligned}$$

Note that sum inside the expectation is simply another expectation, as in the definition of entropy — but since we only want to apply Jensen's inequality on the outer expectation we spell this one out explicitly. Moreover, by definition of the conditional pmf we have $\mathbb{E}[p_{y|X}] = \sum_x p_x p_{y|x} = \sum_x p_{xy} = p_y$. Hence, using concavity of the entropy as a function of the pmf and Jensen's inequality for the outer expectation, we find

$$\begin{aligned} H(Y|X) &= \mathbb{E}\left[\sum_y p_{y|X} \log \frac{1}{p_{y|X}}\right] && (1.31)\\ &\leq \sum_y \Big(\mathbb{E}[p_{y|X}]\Big) \log \frac{1}{\mathbb{E}[p_{y|X}]} && (1.32)\\ &= \sum_y p_y \log \frac{1}{p_y} && (1.33)\\ &= H(Y)\,. && (1.34) \end{aligned}$$

Equality in Jensen's inequality only holds if either $X$ is deterministic or if $p_{y|x} = p_y$ for all $x$ and $y$, but this only holds if $X$ and $Y$ are in fact independent. □

The second relation in Eq. (1.27) can be strengthened by considering three random variables $X$, $Y$ and $Z$. In that case, we have

$$H(X|YZ) \leq H(X|Z) \quad \text{and} \quad H(X|Z) + H(Y|Z) \geq H(XY|Z)\,. \tag{1.35}$$

This is sometimes referred to as *strong sub-additivity*. The proof follows from (regular) sub-additivity, applied to the entropies $H(X|Y, Z = z)$ and $H(X|Z = z)$, and averaging the resulting inequalities.

Show that the two inequalities in (1.35) are equivalent

## 1.3 Mutual information and conditional mutual information

### 1.3.1 Mutual information

We have already established that $H(XY) \neq H(X) + H(Y)$ in general, and hence also $H(Y|X) \neq H(Y)$ by the chain rule. The difference between these two quantities clearly tells us something about how much the uncertainty about $Y$ changes when we learn $X$, or in other words, about how much information $X$ contains about $Y$. This leads us to the definition of mutual information,

The *mutual information* between $X$ and $Y$ is defined as

$$I(X : Y) := H(Y) - H(Y|X) \tag{1.36}$$

It is not evident immediately from the way we defined it here but this expression is symmetric between $X$ and $Y$. Namely, using the chain rule for conditional entropy (recall Proposition 1.3) twice, we can write

$$\begin{aligned} I(X : Y) = H(Y) - H(Y|X) &= H(Y) + H(X) - H(XY) \\ &= H(X) - H(X|Y)\,. \end{aligned} \tag{1.37}$$

The mutual information is thus a symmetric measure of the correlation between the two random variables.

Using these various equivalent expressions it is then easy to derive some bounds on the mutual information. First, sub-additivity of the entropy directly implies that $I(X : Y) \geq 0$, so the mutual information is non-negative, and it vanishes iff the two random variables are independent (a consequence of Proposition 1.4). This is consistent with our intuitive notion of information — we cannot know less than nothing after all! We also cannot know more than everything, i.e. the mutual information can never exceed the entropy of any of its constituent parts.

Using the bounds on entropies established in the previous sections, show that

$$I(X : Y) \leq \log \min\{|\mathcal{X}|, |\mathcal{Y}|\}.$$

Give an example that saturates the bound.

**Example.** *Consider two binary random variables $X$ and $Y$ with joint pmf*

$$\begin{aligned} P_{XY}(0,0) = P_{XY}(1,1) &= \frac{1}{4}(1+r), \\ P_{XY}(0,1) = P_{XY}(1,0) &= \frac{1}{4}(1-r) \end{aligned} \tag{1.38}$$

*for $r \in [-1, 1]$. We can compute the mutual information between $X$ and $Y$*

You might have heard of the correlation coefficient in statistics:

$$\rho = \frac{\mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}}.$$

Can you determine $\rho$ as a function of $r$?

*as follows:*

$$I(X:Y) = H(X) - H(X|Y) = 1 - h\left(\frac{1+r}{2}\right) \tag{1.39}$$

*So this function takes its maximum value at $r = -1$ and $r = 1$ and drops to zero for $r = 0$.*

### 1.3.2 Conditional mutual information

If we have three random variables $X$, $Y$ and $Z$ we can ask for the mutual information between $X$ and $Y$ conditioned on knowing $Z$, the *conditional mutual information*. It is defined as

$$I(X:Y|Z) := \sum_z P_Z(z) I(X:Y|Z=z)\,. \tag{1.40}$$

Various equivalent expressions can then be readily derived, e.g.,

$$I(X:Y|Z) = H(Y|Z) - H(Y|XZ) = H(X|Z) - H(X|YZ)\,. \tag{1.41}$$

Verify (1.41) and (1.42) using the definition in (1.40).

Moreover, the *chain rule* for the mutual information states that

$$I(X:YZ) = I(X:Z) + I(X:Y|Z)\,, \tag{1.42}$$

To show (1.42), you can, for example, use that

$$\begin{aligned} I(X:Y|Z) &= H(X|Z) - H(X|YZ) \\ &= -(H(X) - H(X|Z)) \\ &\quad + (H(X) - H(X|YZ)) \\ &= -I(X:Z) + I(X:YZ). \end{aligned}$$

which can be verified by a close inspection of the definition of both conditional and unconditional mutual information.

On the one hand, consider the case where $X - Z - Y$ form a Markov chain. In this case $P_{X|YZ} = P_{X|Z}$ and thus $H(X|YZ) = H(X|Z)$. As a consequence, the conditional mutual information $I(X:Y|Z)$ as written in (1.41) vanishes. On the other hand, if $I(X:Y|Z) = 0$ then we must have that $I(X:Y|Z=z) = 0$ for every $z$ with $P_Z(z) > 0$ according to our definition in (1.40). Hence, $P_{XY|Z=z} = P_{X|Z=z}P_{Y|Z=z}$ is independent according to Proposition 1.4. This means that $X - Z - Y$ must be a Markov chain. We summarise this in the following proposition:

**Proposition 1.5.** *The following two conditions are equivalent: a) $X - Z - Y$ form a Markov chain, and b) $I(X:Y|Z) = 0$.*

### 1.3.3 Data-Processing inequality

One of the most intriguing properties of the mutual information is the *data-processing inequality* (DPI) for mutual information. It states that the mutual information can never increase when we apply an operation that only acts on one of the parts. Intuitively this tells us that by manipulating one of the random variables without looking at the other we cannot increase the correlations between the pair.

We can formalise this using the notion of Markov chains.

Recall that for any random variables $X$ and $Y$ and any channel $W$ from $Y$ to $Z$, the resulting random variables form a Markov chain $X - Y - Z$.

**Proposition 1.6** (DPI for Mutual Information). *Let $X - Y - Z$ be a Markov chain. Then, $I(X:Y) \geq I(X:Z)$.*

*Proof.* Since $I(X:Z|Y) = 0$, the chain rule for mutual information implies that $I(X:Y) = I(X:YZ)$. It thus remains to show that

$$I(X:Z) \leq I(X:YZ)\,. \tag{1.43}$$

But, since $I(X:Z) = H(X) - H(X|Z)$ and $I(X:YZ) = H(X) - H(X|YZ)$, the relation in Eq. (1.43) is equivalent to the condition $H(X|Z) \geq H(X|YZ)$, which is in turn ensured by the strong subadditivity of entropy. □

Can you also show that

$$I(Y:Z) \geq I(X:Z)$$

under the same assumption?

## 1.4 Relative entropy

### 1.4.1 Log-likelihood ratio and relative entropy

The relative entropy, often referred to as Kullback-Leibler divergence, emerges when we want to compare two different probability distributions. We define it here only for discrete random variables (or rather the respective pmfs). This can be generalised to general probability measures using the notion of Radon-Nikodyn derivatives, but this is outside our scope.

Let $P$ and $Q$ be two pmfs on an alphabet $\mathcal{X}$. The *relative entropy* of $P$ with regards to $Q$ is defined as

$$D(P\|Q) := \sum_{\substack{x\in\mathcal{X}\\ P(x)>0}} P(x)\log\frac{P(x)}{Q(x)}\,. \tag{1.44}$$

if $P(x) > 0 \implies Q(x) > 0$ for all $x \in \mathcal{X}$, and as $D(P\|Q) = +\infty$ otherwise.

In the following, instead of restricting the sum, we will use the convention that $0\log\frac{0}{0} = 0$.

We can alternatively see the relative entropy as the expectation value of the *log-likelihood ratio* (LLR), namely we can write

$$D(P\|Q) = \mathbb{E}\left[Z(X)\right], \qquad \text{where} \qquad Z(X) = \log\frac{P(X)}{Q(X)} \tag{1.45}$$

and $X$ is distributed according to $P$. The random variable $Z(X)$ is the LLR. It takes on the role of the surprisal in the definition of entropy. If the LLR is large for events that have high probability under $P$ then $P$ and $Q$ are easy to distinguish and the relative entropy is large. We will explore this random variable and its distribution much more

**Example.** Consider two Bernoulli distributions $P = \text{Bern}(p)$ and $Q = \text{Bern}(q)$. Then the relative entropy evaluates to

$$D(P\|Q) = p\log\frac{p}{q} + (1-p)\frac{1-p}{1-q}.$$

What values does the random variable $Z$ take in the above example, and with what probability?

when we discuss the information spectrum method and hypothesis testing later on.

Just by manipulating the definition, we are able to show the following equivalences.

**Proposition 1.7.** *Let $X$ and $Y$ be random variables on alphabets $\mathcal{X}$ and $\mathcal{Y}$. Moreover, let $U$ be a uniform random variable on $\mathcal{X}$. Then the following relations are true:*

$$H(X) = \log|\mathcal{X}| - D(P_X\|U_X) \tag{1.46}$$
$$H(X|Y) = \log|\mathcal{X}| - D(P_{XY}\|U_X \times P_Y) \tag{1.47}$$
$$I(X:Y) = D(P_{XY}\|P_X \times P_Y)\,. \tag{1.48}$$

You will prove these equivalences in Exercise 1.5. They turn out to be very useful because they essentially tell us that once we established properties of the relative entropy this has immediate consequences also for the derived quantities.

Can you express the conditional mutual information $I(X{:}Z|Y)$ in terms of the relative entropy?

### 1.4.2 *Positivity*

We will need two important properties of the relative entropy. The first proposition establishes that the relative entropy is always positive.

**Proposition 1.8.** *For any two pmfs $P$ and $Q$, we have $D(P\|Q) \geq 0$ with equality iff $P = Q$.*

*Proof.* We can assume without loss of generality that the quantity is finite, as otherwise the statement is trivially true. We first note that $x \mapsto -\log x$ is strictly convex. Hence,

$$D(P\|Q) = \sum_{x:P(x)>0} P(x)\log\frac{P(x)}{Q(x)} \tag{1.49}$$
$$= \sum_{x:P(x)>0} P(x)\left(-\log\frac{Q(x)}{P(x)}\right) \tag{1.50}$$
$$\geq -\log\left(\sum_{x:P(x)>0} P(x)\frac{Q(x)}{P(x)}\right) \tag{1.51}$$
$$= -\log\left(\sum_{x:P(x)>0} Q(x)\right) \tag{1.52}$$
$$\geq -\log\left(\sum_x Q(x)\right) = 0\,. \tag{1.53}$$

Equality in the second inequality only holds if $P$ and $Q$ have the same support. Moreover, equality in the first inequality holds if

$\frac{Q(x)}{P(x)}$ is independent of $x$ for any $x$ in the support of $P$. These two statements are both true only if $P(x) = Q(x)$ for all $x \in \mathcal{X}$, and thus $P = Q$. □

An immediate corollary of Propositions 1.7 and 1.8 is that $I(X:Y)$ is positive and zero iff $X$ and $Y$ are independent.

Can you prove this?

### 1.4.3 *Data-processing inequality*

Finally, there is one property of the relative entropy that, in conjunction with Proposition 1.7, implies most other properties of entropy, conditional entropy and mutual information. It states that applying a noisy operation, i.e., a stochastic map or channel, on both arguments of the relative entropy will never increase it. Together with the positivity of relative entropy this justifies that we think of it as a measure of similarity or distinguishability. If the relative entropy is small the two pmfs are similar and hard to distinguish by observing the outcomes of a random experiment. Observing the outcomes after further noise has been applied should make distinguishing them even harder, and that is exactly what the *data-processing inequality* for relative entropy tells us.

**Proposition 1.9** (DPI for Relative Entropy)**.** *Let $P_X$ and $Q_X$ be two pmfs on an alphabet $\mathcal{X}$ (the input distributions), and let $W_{Y|X}$ be a channel (a conditional pmf). Define the marginals (the output distributions)*

$$P_Y(y) = \sum_{x\in\mathcal{X}} W_{Y|X}(y|x)P_X(x) \qquad \text{and}$$
$$Q_Y(y) = \sum_{x\in\mathcal{X}} W_{Y|X}(y|x)Q_X(x)\,. \tag{1.54}$$

*Then, the* data-processing inequality *(DPI) states that*

$$D(P_X\|Q_X) \geq D(P_Y\|Q_Y)\,. \tag{1.55}$$

*Proof.* Consider the joint distributions $P_{XY}(x,y) = W_{Y|X}(y|x)P_X(x)$ and $Q_{XY}(x,y) = W_{Y|X}(y|x)Q_X(x)$, using the usual shorthand notation for conditional and marginal distributions. We first show that

$$D(P_{XY}\|Q_{XY}) - D(P_Y\|Q_Y) = \sum_{x,y} p_{xy}\log\frac{p_{xy}}{q_{xy}} - \sum_y p_y\log\frac{p_y}{q_y} \tag{1.56}$$
$$= \sum_{x,y} p_{xy}\left(\log\frac{p_{xy}}{q_{xy}} - \log\frac{p_y}{q_y}\right) \tag{1.57}$$
$$= \sum_y p_y \sum_x p_{x|y}\log\frac{p_{x|y}}{q_{x|y}} \tag{1.58}$$
$$= \sum_y p_y\, D(P_{X|Y=y}\|Q_{X|Y=y}) \geq 0\,, \tag{1.59}$$

where we have used the positivity of relative entropy in the last step. Similarly, we have

$$D(P_{XY}\|Q_{XY}) - D(P_X\|Q_X) = \sum_x p_x\, D(P_{Y|X=x}\|Q_{Y|X=x}) = 0 \quad (1.60)$$

since $Q_{Y|X} = P_{Y|X} = W_{Y|X}$ by construction of the joint distribution. Combining Eqs. (1.56)–(1.59) and (1.60) yields the desired inequality. □

It turns out that all the properties of entropy, conditional entropy and mutual information we discussed previously can be derived form the DPI. As an example we give here a strengthening of the strong sub-additivity, which we call the data-processing inequality for conditional entropy. It intuitively states that any processing of the side information can at most increase the conditional entropy.

**Corollary 1.10** (DPI for conditional entropy)**.** *Let $P_{XY}$ be a joint pmf and $W_{Z|Y}$ a channel. Define*

$$P_{XZ}(x,z) = \sum_y P_{XY}(x,y) W_{Z|Y}(z|y)\,. \quad (1.61)$$

*Then, we have $H(X|Y) \le H(X|Z)$.*

Can you derive a DPI for mutual information using the same idea? Compare with Proposition 1.6.

*Proof.* Let us express the inequality in terms of relative entropies using Proposition 1.7. This reads

$$\log|\mathcal{X}| - D(P_{XY}\|U_X \times P_Y) \le \log|\mathcal{X}| - D(P_{XZ}\|U_X \times P_Z)\,. \quad (1.62)$$

or simply $D(P_{XY}\|U_X \times P_Y) \ge D(P_{XZ}\|U_X \times P_Z)$. But this is imply the DPI applied to the channel $W_{Z|Y}$ that happens to leave $X$ untouched. □

## 1.5 Exercises

**Exercise 1.1.** *Adapt the proof of Proposition 1.2 showing that $H(X) \le \log|\mathcal{X}|$ for binary random variables so that it works for general finite alphabets $\mathcal{X}$.*

**Exercise 1.2.** *For each item, find an example of random variables $X$, $Y$ and $Z$ such that the desired relations holds:*

**Hint:** An alphabet of one or two bits for each random variable will suffice.

a) $H(X|YZ) = 0$ *and* $H(X|Y) = H(X|Z) = 1$.

b) $I(X:Y|Z) = 1$ *and* $I(X:Y) = 0$.

c) $I(X:Y) = 1$ *and* $I(X:Y|Z) = 0$.

d) $I(X:Y) = I(X:Z) = 1$ *and* $I(Y:Z) = 0$.

**Exercise 1.3.** *Compute the mutual information $I(X:Y)$ for the pmf*

| $P_{XY}$ | $x=0$ | $x=1$ | $x=2$ |
|---|---|---|---|
| $y=0$ | $\frac{1}{2}$ | $0$ | $0$ |
| $y=1$ | $0$ | $\frac{1}{4}$ | $\frac{1}{4}$ |

**Exercise 1.4.** *Consider two sequences of random variables $X_1,\dots,X_n$ and $Y_1,\dots,Y_n$. Show that if $X_1,\dots,X_n$ are mutually independent, then*

$$I(X_1,\dots,X_n:Y_1,\dots,Y_n) \geq \sum_{i=1}^{n} I(X_i:Y_i)\,.$$

*On other hand, if given $Y_i$ the random variable $X_i$ is conditionally independent of all the other rvs for all $i=1,\dots,n$, then show that*

$$I(X_1,\dots,X_n:Y_1,\dots,Y_n) \leq \sum_{i=1}^{n} I(X_i:Y_i)\,.$$

**Exercise 1.5** (Parent quantity)**.** *Let $X$ and $Y$ be random variables on alphabets $\mathcal{X}$ and $\mathcal{Y}$ with joint pmf $P_{XY}$. Moreover, let $U$ be a uniform random variable on $\mathcal{X}$. Show the following relations:*

a) $H(X) = \log|\mathcal{X}| - D(P_X\|U_X)$.

b) $H(X|Y) = \log|\mathcal{X}| - D(P_{XY}\|U_X \times P_Y)$.

c) $I(X:Y) = D(P_{XY}\|P_X \times P_Y)$.

**Exercise 1.6** (Information Spectrum)**.** *Given a random variable $X$ governed by the pmf $P$ or an alternative pmf $Q$, recall that the log-likelihood ratio is defined as the random variable $Z(X) = \log\frac{P(X)}{Q(X)}$ where $X$ is distributed according to $P$.*

a) *Give an expression for* $\mathrm{Var}(Z)$*. This quantity is called the relative entropy variance and denoted by $V(P\|Q)$.*

*Consider now a sequence of i.i.d. random variables $X^n = (X_1, X_2, \dots, X_n)$ on $\mathcal{X}^n$ where each $X_i$ is governed by the pmf $P$ or an alternative pmf $Q$. We are interested in pmf of the log-likelihood ratio $Z(X^n)$.*

b) *Show that $Z(X^n) = \sum_{i=1}^n Z(X_i)$. What is $\mathbb{E}[Z(X^n)]$ and $\mathrm{Var}\big(Z(X^n)\big)$?*

c) *Let us now consider the quantity $P[Z(X^n) \leq nR]$ in the limit of large $n$ for different values of $R$. Show that*

**Hint:** Argue using the weak law of large numbers.

$$\lim_{n\to\infty} P[Z(X^n) \leq nR] = \begin{cases} 0 & \text{if} \quad R < D(P\|Q) \\ 1 & \text{if} \quad R > D(P\|Q) \end{cases}.$$

d) *In Chapter 3 we will encounter the quantity*

**Hint:** Verify that $\frac{1}{n}D_s^\epsilon(P^n\|Q^n) = \sup\{k\in\mathbb{R} : P[\frac{1}{n}Z(X^n) \leq k] \leq \epsilon\}$.

$$D_s^\epsilon(P^n\|Q^n) := \sup\{k\in\mathbb{R} : P[Z(X^n)\leq k] \leq \epsilon\},$$

*which, in words, is asking the largest k such that the tail of the distribution of Z that lies below k has cumulative probability at most $\epsilon$. Show that $D_s^\epsilon(P^n\|Q^n) = nD(P\|Q) + o(n)$, or equivalently,*

$$\lim_{n\to\infty} \frac{1}{n} D_s^\epsilon(P^n\|Q^n) = D(P\|Q)\,.$$

e) *Optional: Show that in the next order in n, we have*

$$D_s^\epsilon(P^n\|Q^n) = nD(P\|Q) + \sqrt{nV(P\|Q)}\,\Phi^{-1}(\epsilon) + o\left(\sqrt{n}\right).$$

**Hint:** The statement can be shown using the central limit theorem.

# 2
# *Source coding*

**Intended learning outcomes:**

- You can determine if a code is instantaneous and if the codeword lengths are optimal using the McMillan-Kraft inequality.
- You can evaluate the quality of a variable-length code by comparing it to the fundamental limits.
- You can construct a Huffman code for any discrete source and understand the algorithm's properties.
- You understand the mathematical model used to study block codes asymptotically. You know what rates are and what achievability, converse and strong converse mean in the context of source coding.
- You understand Fano's inequality and how it can be used to show a converse.
- You can work with typical sets and how they can be used to show achievable rates.

**Book reference:** Chapter 5 in Cover & Thomas[1].

[1] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991. ISBN 9780471748823. DOI: 10.1002/047174882X

## 2.1 *Problem setup and definitions*

In this chapter we are concerned with removing redundancy. In the first section we will introduce the formal mathematical model we use to investigate source coding, or compression.

### 2.1.1 *Data source*

We are given data as a sequence of symbols, for example these could be numbers, letters, colours of pixels, etc., and we would like to store that data in a (preferably short) sequence of bits. (We could generalise to larger alphabets, but conceptually nothing changes so we restrict ourselves to bits here to simplify presentation.) We start by formally defining what we mean by a *data source*, or simply *source* in the remainder of this chapter.

A *data source* is an infinite sequence of random variables

$$\boldsymbol{X} = X_1, X_2, \ldots, X_k, \ldots . \tag{2.1}$$

- A source is called *discrete* if the random variables are discrete, i.e. if the source outputs in each iteration $i \in \mathbb{N}$ values from a finite set $\mathcal{X}$.
- A source is furthermore called *memoryless* if the $X_i$ are independent and identically distributed (i.i.d.) according to the same pmf $P_X$, i.e., if we have

$$P_{X_1X_2\ldots X_k\ldots}(x_1, x_2, \ldots, x_k, \ldots) = P_X(x_1)P_X(x_2)\ldots P_X(x_k)\ldots. \tag{2.2}$$

Memoryless here refers to the fact that the distribution of $X_i$ does not depend on the value of $X_{i-1}$ or any other symbol in the sequence, or, formally, $P_{X_i|X_1X_2\ldots X_{i-1}} = P_{X_i} = P_X$. We will not consider sources that output continuous values in this module.

For most of our theoretical analysis, we will consider a *discrete memoryless source (DMS)*. An example of such a source is the sequence of face values one gets by throwing the same (fair or unfair) die repeatedly. Generally, the assumption that a source is memoryless is simplifying the analysis but in fact most sources do not satisfy this exactly. For example, think of a book (written in English) as a sequence of letters and a source reproducing them one by one. If $X_{i-1} =$ 'q', then $X_i =$ 'u' with much higher probability than the frequency of 'u' in English text would otherwise suggest. Hence, this source is far from memoryless and the corresponding distribution of the random variable is not i.i.d.. Nonetheless, understanding the simple case of discrete memoryless sources properly will allow us to get an intuition for more loosely structured sources as well. Various more complicated models of sources have been analysed in the literature.

Can you see why we cannot expect to store and perfectly recover a continuous variable using finite (digital) memory?

### 2.1.2 *Source codes*

Next we introduce *source codes*, or simply *codes* for the remainder of this chapter.

A *source code* is a function $C$ that maps outputs of the source $x \in \mathcal{X}$ to bit strings of variable length, $C(x) \in \{0,1\}^*$. We denote by $\ell(x)$ the length of the *codeword* $C(x)$.

- A code is called a *fixed-length* code if $\ell(x) = \ell$ is constant, otherwise it is called a *variable-length* code.

(N) A code is called *non-singular* if $C$ is injective, i.e. if every $x \in \mathcal{X}$ is mapped to a unique bit string.

We say that a bit string is a prefix of another bit string if the latter starts with the former. For example, 01 is a prefix of 0100 and 1, 11 and 110 are all prefixes of the string 110101.

(P) A code is called a *prefix-free code* if for any pair $x, x' \in \mathcal{X}$ with $x \neq x'$, the codeword $C(x)$ is not a prefix of the codeword $C(x')$.

(U) A code is *uniquely decodable* if there exists a decoder that, for any $n \in \mathbb{N}$ and any sequence $x^n \in \mathcal{X}^n$, can uniquely recover $x^n$ from the bit string $C(x_1)C(x_2)\dots C(x_n)$.

(I) A code is *instantaneous* if it is uniquely decodable and if the decoder can deduce the $k$-th symbol $x_k$ as soon it has seen the bit string $C(x_1)C(x_2)\dots C(x_k)C(x_{k+1})\dots$ up to and including all of $C(x_k)$, even if there is no guarantee that the string is complete.

Let us note that the codes we consider here are not as general as they could be. In fact, we could also consider codes that take a variable length sequence of input symbols to a codeword (of either fixed or variable length). Such codes are in fact often used in practical applications. A prominent example is the Lempel–Ziv–Markov algorithm for lossless compression, which uses a dictionary to replace often reoccurring variable-length sequences with shorter codewords.

Let us now discuss some of the interrelations between all these code properties. Clearly, any instantaneous code is uniquely decodable and every uniquely decodable code is non-singular since, whenever two symbols are mapped to the same string, it is impossible to invert this map in a unique way. However, we observe that not every non-singular code is uniquely decodable. Consider a code on $\mathcal{X} = \{0, 1, 2, 3\}$ that yields the binary representation

$$C(0) = 0, \quad C(1) = 1, \quad C(2) = 10, \quad C(3) = 11\,. \tag{2.3}$$

This code is non-singular but not a prefix-free code. The codeword string 110 could either be produced by the source sequence $(3, 0)$, by $(1, 2)$ or even by $(1, 1, 0)$, so there is no way for a decoder to distinguish between these three cases.

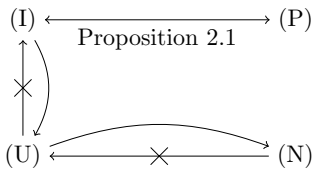



Figure 2.1: **Code properties.** Relationships between the different code properties of variable-length source codes. Recall that (I) stands for instantaneous, (P) for prefix-free, (U) for uniquely decodable and (N) for non-singular.

**Proposition 2.1.** *A code is instantaneous if and only if it is prefix-free.*

*Proof.* We first show that a prefix-free code is instantaneous by constructing a decoder. The decoder will read the sequence $C(x_1)C(x_2)\dots$ bit by bit. Once $C(x_1)$ is fully read we can immediately deduce that the first source symbol was $x_1$ since $C(x_1)$ cannot be a prefix for a longer codeword. Similarly, with this rule in mind, since there is no other codeword that is a prefix to $C(x_1)$ we can be assured that this is indeed the first symbol we will decode. The same procedure continues for the remainder of the string with $C(x_2)C(x_3)\dots$. We know where every codeword ends and can decode them individually and instantaneously.

To verify the other direction, simply note that if a decoder can decide instantly once it has seen the codeword $C(x)$ this implies that $C(x)$ cannot be a prefix to any other codeword $C(x')$. Since this is true for any $x \neq x'$, the code must be a prefix-free code. □

Thus, in particular, any prefix-free code is uniquely decodable. However, the converse is not true in general, i.e. not every uniquely decodable code is a prefix-free code. Consider as an example the code

$$C(\text{'a'}) = 1, \quad C(\text{'b'}) = 10, \quad C(\text{'c'}) = 00\,. \tag{2.4}$$

After seeing 10 we cannot decide if the first symbol was 'a' or 'b' even though we have seen the full codeword of the first symbol, hence this code is not instantaneous. However, once we have seen a full sequence of codewords, we can decode uniquely by looking at the parity of the number of 0's between two 1's. These relationships are summarised in Figure 2.1.

For example, if we see 1001 this decodes to 'aca' whereas 10001 would decode to 'bca'.

Can you come up with a formal decoder for this code?

Codes can be conveniently represented by binary trees. Binary trees are connected graphs without cycles (trees) where each node (except the root) has exactly one parent and either zero (in which case it is called a leaf) or at most two children. The branches emanating from the root and each node are assigned values 0 or 1 and codewords are composed by following a path from the root to a node.

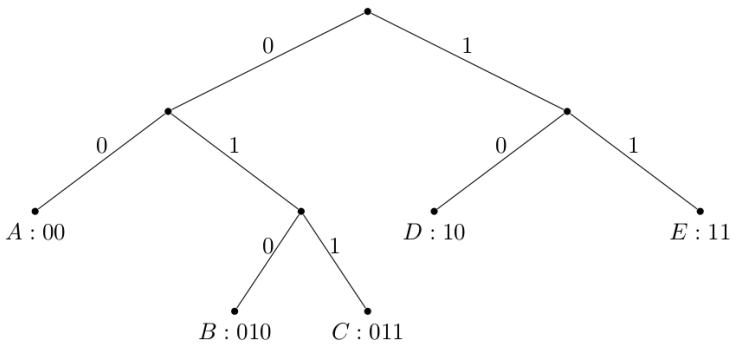



Figure 2.2: **Example of a code tree.** The codewords at the leaves are composed of the symbols associated with the edges on the path from the root to the leaf.

The codeword length is equivalent to the depth (i.e. the distance from the root) of the node in the three. For a fixed-length code all the codewords are at the same depth (or level) of the tree.

Can you see why trees for prefix-free codes have this property?

A code is a prefix-free code if and only if on every path from a leaf to the root there is at most one codeword.

The next two sections will be devoted to variable-length and fixed-length block codes (defined later), respectively.

## 2.2 *Variable-length codes*

Before we discuss particular codes, we first want to establish some fundamental limits all codes need to satisfy.

### 2.2.1 *Optimal codeword lengths*

The Kraft inequality gives a lower bound on the lengths of codewords in any instantaneous code, and it is the first fundamental limit we will establish. It shows us that no code with shorter codeword lengths can exist, and thus if a code achieves equality in (2.5) we know it is optimal in this regard. We present a slightly more general result, the MacMillan–Kraft inequality, which applies for any uniquely decodable code (and not just prefix-free codes).

**Proposition 2.2** (McMillan–Kraft inequality)**.** *Any uniquely decodable code must satisfy the inequality*

$$\sum_{x \in \mathcal{X}} 2^{-\ell(x)} \leq 1 \,. \tag{2.5}$$

*Conversely, given a set of codewords lengths satisfying Eq. (2.5), it is possible to construct a prefix-free code with these lengths.*

We show the inequality only for prefix-free codes, and the general proof for uniquely decodable (not necessarily prefix-free) codes will be covered in Exercise 2.1.

*Proof.* We use the one-to-one correspondence of prefix-free codes with binary trees for which the codewords are leaves. For any tree we may assign to every node a value $2^{-d}$ where $d$ is the depth in the tree. The root thus gets the value 1. To show the Kraft inequality we simply need to show that the sum of all the leaf values in a tree cannot exceed 1. To see that this is correct, simply note that by our construction any parent node's value is simply the sum of its children's values, so as we build up the binary tree from the root the sum of the values on all leaves is always exactly 1.

Conversely, given a set of codeword lengths satisfying the inequality, we can always create a binary tree with leaves at the corresponding depths and populate the leaves with codewords. If the inequality is strict there will be unused leaves in the tree. □

### 2.2.2 *Optimal expected codeword length*

Finding codewords with short lengths is however only half of the problem — we also need to assign those codewords to elements of $\mathcal{X}$.

And we want to do this in such a way as to minimise the expected length of the codeword. That is, for a code $C(x)$ with codeword lengths $\ell(x)$, we define the *expected codeword length* as

$$\bar{\ell}(C) := \sum_x P_X(x)\ell(x) = \mathbb{E}[\ell(X)] \tag{2.6}$$

Using the McMillan–Kraft inequality from Proposition 2.2, we can lower bound $\bar{\ell}(C)$ with the entropy $H(X)$ for any uniquely decodable code.

**Proposition 2.3.** *For any uniquely decodable code $C$ for a discrete source $X$ with distribution $P_X$, we have*

$$\bar{\ell}(C) \geq H(X)\,. \tag{2.7}$$

*Moreover, the equality is saturated if only if the codeword lengths saturate the McMillan-Kraft inequality and $P_X(x) = 2^{-\ell_x}$ for some $\ell_x \in \mathbb{N}$.*

*Proof.* We evaluate

$$\begin{aligned}
\bar{\ell}(C) - H(X) &= \mathbb{E}\left[\ell(X) - \log\frac{1}{P_X(X)}\right] && (2.8)\\
&= \mathbb{E}\left[\log\frac{P_X(X)}{2^{-\ell(X)}}\right] && (2.9)\\
&\geq \mathbb{E}\left[\log\frac{t\cdot P_X(X)}{2^{-\ell(X)}}\right] && (2.10)\\
&= D(P_X\|Q_X) && (2.11)\\
&\geq 0, && (2.12)
\end{aligned}$$

where we introduced the constant $t = \sum_x 2^{-\ell(x)} \leq 1$ (by the McMillan–Kraft inequality) and the pmf $Q_X(x) = \frac{1}{t}2^{-\ell(x)}$. The final inequality is simply due to the non-negativity of relative entropy.

If the conditions for saturation are met we can see that the two inequalities become equalities as $t = 1$ and $P_X = Q_X$ if we choose $\ell(x) = \ell_x$. Conversely, using the strict monotonicity of the logarithm and the positive definiteness of the relative entropy, we see that these conditions are in fact necessary to achieve equality. □

### 2.2.3 *Shannon code*

The above result allows us to show that certain codes have optimal expected codeword lengths. For example for the source $X$ that outputs symbols 'a', 'b' and 'c' with probabilities $\frac{1}{2}$, $\frac{1}{4}$ and $\frac{1}{4}$, respectively, the code

$$C(\text{'a'}) = 0, \quad C(\text{'b'}) = 10, \quad C(\text{'c'}) = 11 \tag{2.13}$$

satisfies $\bar{\ell}(C) = \frac{1}{2} + 2 \cdot \frac{1}{4} \cdot 2 = \frac{3}{2}$ and $H(X) = \frac{1}{2}\log 2 + 2 \cdot \frac{1}{4}\log 4 = \frac{3}{2}$, and thus, we know that it is optimal thanks to Proposition 2.3.

The above example is constructed in such a way that all the probabilities are negative powers of 2 and the codeword lengths satisfy the Kraft inequality with equality, and in this particular case it is easy to see from the proof of Proposition 2.3 that $\bar{\ell}(C) = H(X)$. If the probabilities do not have this form the same construction does not generally work. However, we can show the following.

**Proposition 2.4** (Shannon Code). *For any discrete source $X$ with distribution $P_X$ there exists a prefix-free code $C$ with $\bar{\ell}(C) < H(X) + 1$.*

The code we construct to proof this is called the Shannon code, and it not optimal in general. However, this bound, together with Proposition 2.3, shows that we lose at most 1 bit per symbol using this code.

*Proof.* We can choose codeword lengths $\ell(x) = \left\lceil \log \frac{1}{P_X(x)} \right\rceil$. These satisfy the Kraft inequality since

$$\sum_x 2^{-\ell(x)} = \sum_x 2^{-\left\lceil \log \frac{1}{P_X(x)} \right\rceil} \leq \sum_x 2^{-\log \frac{1}{P_X(x)}} = \sum_x P_X(x) = 1\,. \tag{2.14}$$

Hence, using Proposition 2.2 we may construct a prefix-free codes using these lengths. Moreover, for this code we have

$$\begin{aligned} \bar{\ell}(C) &= \sum_x P_X(x) \left\lceil \log \frac{1}{P_X(x)} \right\rceil && (2.15)\\ &< \sum_x P_X(x) \left( \log \frac{1}{P_X(x)} + 1 \right) && (2.16)\\ &= H(X) + 1\,. && (2.17) \end{aligned}$$

□

For a more explicit construction of the Shannon code, see also Exercise 2.3.

### 2.2.4 *Huffman codes*

In this section we will construct prefix-free codes with optimal expected codeword length, so-called Huffman codes[2]. We will first learn how to construct the codes and then use this construction to show optimality.

In 1951, David Huffman and his MIT information theory classmates were given the choice of a term paper or a final exam. The professor, Robert Fano, assigned a term paper on the problem of finding the most

[2] David A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952. DOI: 10.1109/JRPROC.1952.273898

efficient binary code. Huffman, unable to prove any codes were the most efficient, was about to give up and start studying for the final when he hit upon the idea of using a frequency-sorted binary tree and quickly proved this method the most efficient. In doing so, Huffman outdid Fano, who had worked with information theory inventor Claude Shannon to develop a similar code.

The code is constructed using Algorithm 2.1, which step-by-step merges a forest of trivial binary trees into a single binary tree.

**Input:** List of symbols $x \in \mathcal{X}$ with probabilities $p_x = P_X(x)$
**Output:** Binary tree for the Huffman code
% initialise forest
**for** *each* $x \in \mathcal{X}$ **do**
  Create a tree with a root node labelled by the probability $p_x$ (and the symbol $x$) and no other nodes;
  Add this tree to the forest;
**end**
% condense forest into a single tree
**while** *number of trees in the forest is larger than* 1 **do**
  select two trees whose roots have smallest probabilities, $p$ and $p'$;
  join the two trees by adding a new root with probability $p + p'$ and the two trees as children; the edges are labelled with 0 and 1;
**end**
return last remaining tree in the forest;

**Algorithm 2.1:** Construction of a Huffman code tree.

The construction is not unique because in each step we can assign the labels '0' and '1' in either way to the two children. Moreover, we are asked to select the two trees with smallest probabilities, but there might be different such pairs, e.g. if we start with a source $X$ with symbols and probabilities ('a', $\frac{1}{3}$), ('b', $\frac{1}{3}$), ('c', $\frac{1}{6}$), ('d', $\frac{1}{6}$) then both of these codes, $C_1$ and $C_2$, are possible Huffman codes:

Can you retrace how these codes are created step-by-step?

$$C_1(\text{'a'}) = 00, \quad C_1(\text{'b'}) = 01, \quad C_1(\text{'c'}) = 10, \quad C_1(\text{'d'}) = 11 \tag{2.18}$$
$$C_2(\text{'a'}) = 0, \quad C_2(\text{'b'}) = 10, \quad C_2(\text{'c'}) = 110, \quad C_2(\text{'d'}) = 111 \tag{2.19}$$

The codeword lengths for both codes are optimal according to the Kraft inequality, that is, we have

$$\sum_x 2^{-\ell(x)} = 4 \cdot 2^{-2} = 1 \qquad \text{and} \tag{2.20}$$
$$\sum_x 2^{-\ell(x)} = 2^{-1} + 2^{-2} + 2 \cdot 2^{-3} = 1 \tag{2.21}$$

for $C_1$ and $C_2$, respectively. We can further compute their respective

expected codeword lengths. This yields

$$\bar{\ell}(C_1) = 2 \tag{2.22}$$

$$\bar{\ell}(C_2) = \frac{1}{3} \cdot 1 + \frac{1}{3} \cdot 2 + \frac{1}{6} \cdot 3 + \frac{1}{6} \cdot 3 = 2 \tag{2.23}$$

Let us compare this to the entropy $H(X) = 2 \cdot \frac{1}{3} \log 3 + 2 \cdot \frac{1}{6} \log 6 \approx 1.92$. So from the entropy bound alone we cannot deduce that these codes are optimal in terms of the expected codeword length — but they in fact are!

**Proposition 2.5** (Huffmann code)**.** *Given a source $X$ with probability distribution $P_X$, a code constructed using Algorithm 2.1 achieves the minimal expected codeword length for any prefix-free code.*

We call such a code with minimal expected length an *optimal (prefix-free) code*. The proof relies on the following lemma, which we show first.

**Lemma 2.6.** *There exists an optimal prefix-free code whose corresponding code tree has the following properties:*

1. *The two longest codewords are siblings.*
2. *Their respective source symbols have the two smallest probabilities (this is not always unique).*

*Proof.* An optimal code always corresponds to a binary tree with no unused leaves — if not we can compress the tree by removing the parent of the unused leaf, reducing the expected codeword length. There is always at least one pair of leaves at maximum depth, and those are thus occupied with codewords.

If those codewords would not correspond to the two symbols with smallest probability we could exchange symbols to move them there, which would decrease the expected codeword length. □

*Proof of Proposition 2.5.* The recursive formulation of the Huffman algorithm in Algorithm 2.2 is useful here. Clearly the algorithm produces an optimal code when $|\mathcal{X}| = 2$ since in this case the optimal code simply assigns the codewords 0 and 1 to the two symbols, and this is exactly what the output of the Huffman algorithm will be.

We will thus prove optimality by induction as follows. Let us, without loss of generality, label elements such that our source symbols have probabilities $p_1 \geq p_2 \geq \ldots \geq p_n$ and are ordered such that the Huffman algorithm will pick $p_{n-1}$ and $p_n$ as the smallest elements if there are ambiguities. By induction we may assume

**RecursiveHuffman:**
**Input:** Forest $f_{\text{in}}$
**Output:** Forest $f_{\text{out}}$
**if** *number of trees in* $f_{\text{in}} = 1$ **then**
  return $f_{\text{out}} = f_{\text{in}}$;
**else**
  select two trees in $f_{\text{in}}$ whose roots have smallest probabilities;
  create new forest $f'$ from $f_{\text{in}}$ by joining the selected trees as in Algorithm 2.1;
  return $f_{\text{out}} =$ **RecursiveHuffman**$(f')$;
**end**

**Algorithm 2.2:** Recursive formulation of the Huffman algorithm.

that the recursive Huffman algorithm provides us with an optimal tree when we call it for $n-1$ trees with root probabilities $p_1, p_2, \ldots, p_{n-2}, p_{n-1}+p_n$. We denote the expected codeword length of this optimal tree by $L^*_{n-1}(p_1, p_2, \ldots, p_{n-2}, p_{n-1}+p_n)$.

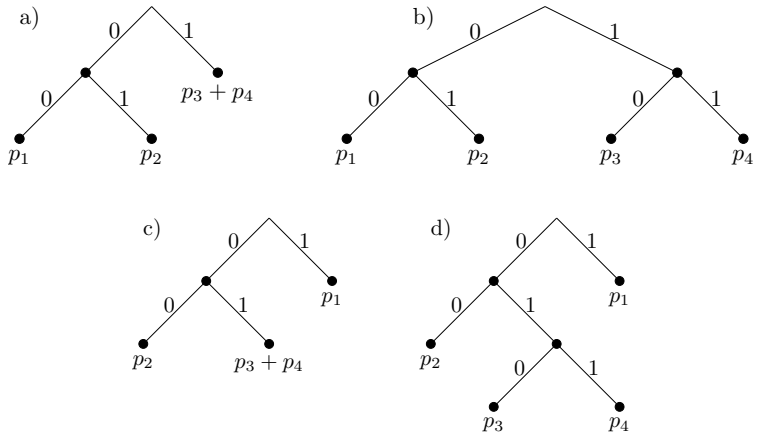



Figure 2.3: **Optimality of Huffman coding with 4 symbols, recursive step.** Assume $p_1 \geq p_2 \geq p_3 \geq p_4$. We are given a minimal length code for $(p_1, p_2, p_3+p_4)$, which must be either of the form a) if $p_3+p_4 \geq p_1$ or of the form c) if $p_3+p_4 \leq p_1$. If equality holds both solutions are optimal. The trees produced by the Huffman algorithm for $(p_1, p_2, p_3, p_4)$ (assuming by induction it produces an optimal tree for $(p_1, p_2, p_3+p_4)$), are thus as in b) and d), respectively. In both cases, we have $L_4 = L^*_3 + p_3 + p_4$.

The Huffman algorithm for $n$ symbols, by definition in its recursive form, will first merge $p_n$ and $p_{n-1}$ and then proceed exactly as the algorithm for $n-1$ symbols discussed above. It produces exactly the same tree but with the the $(n-1)$-th node split into two siblings with probabilities $p_{n-1}$ and $p_n$ (see Figure 2.3 for an example). Its expected codeword length, $L_n$, thus satisfies

$$L_n = L^*_{n-1}(p_1, p_2, \ldots, p_{n-2}, p_{n-1}+p_n) + p_{n-1} + p_n\,. \tag{2.24}$$

Note that we added $p_{n-1}+p_n$ as compared to the tree for $n-1$ symbols those two leaves are now one level deeper, which increases the codeword length by 1 with probability $p_{n-1}+p_n$.

On the other hand, we have

$$L^*_{n-1}(p_1, p_2, \dots, p_{n-2}, p_{n-1} + p_n) \\ \leq L_{n-1} = L^*_n(p_1, p_2, \dots, p_{n-1}, p_n) - p_{n-1} - p_n \,. \tag{2.25}$$

Here the inequality simply states that the optimal expected length is not larger then the expected length $L_{n-1}$ for a specific valid prefix-free code (which just follows from the definition of optimal). We can construct such a prefix-free code for the $n-1$ symbols by taking an optimal prefix-free code for $n$ symbols and merging the two leaves at maximum depth with minimal probability (which exist due to Lemma 2.6) into a single leaf. Such a code has expected codeword length $L^*_n(p_1, p_2, \dots, p_{n-1}, p_n) - p_{n-1} - p_n$ since we decreased the depth by 1 for the two codewords with smallest probability. This yields the equality in (2.25).

Combining Eqs. (2.24) and (2.25) yields

$$L_n \leq L^*_n(p_1, p_2, \dots, p_{n-1}, p_n) \,. \tag{2.26}$$

This proves that the Huffman code construction is optimal. □

## 2.3 *Fixed-length block codes*

Fixed-length codes have the property that all codewords are equally long. If we require error-free compression, then there is not much flexibility: the expected codeword length has to be equal to $\lceil \log |\mathcal{X}| \rceil$ (assuming that every source symbol appears with strictly positive probability). The picture gets dramatically more interesting if we encode a whole block of $n$ source symbols and only require that the probability of a decoding error vanishes as $n \to \infty$.

### 2.3.1 *Setup for block coding*

As we are observing a long sequence of symbols $X_1, X_2, \dots, X_k, \dots$, one thing we can do is to treat a block of, say $n$, symbols as a single symbol (with a much larger alphabet of size $|\mathcal{X}|^n$) and then try to find efficient codes for such blocks. In other words, a block code of length $n$ takes a sequence of $n$ source outputs $x^n \in \mathcal{X}^n$ as input and outputs a binary string. We will formally define it below. Obviously then we can no longer encode and decode instantaneously as we will need to wait for the full block to perform the encoding, and the decoding operation will in turn yield a full block as well.

So for a block code, we observe a sequence of random variables $X^n = (X_1, \dots, X_n)$ from a discrete memoryless source and we would like to compress it into a random variable $M \in \{0,1\}^L$, the codeword, using an encoder, a function $e$ from $X^n$ to $M$. Later on, the decoder

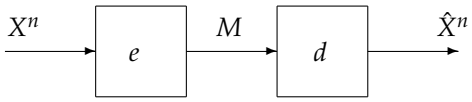



Figure 2.4: Illustration of the fixed-to-fixed length source coding problem.

$d$ will produce an estimate $\hat{X}^n$ of $X^n$ from $M$. See Fig. 2.4 for an illustration. If we demand that

$$P(\hat{X}^n \neq X^n) = 0 \tag{2.27}$$

then $M$ needs to take on at least

$$\left|\{x \in \mathcal{X} : P_X(x) > 0\}\right|^n \tag{2.28}$$

different values, and thus we need to choose

$$L \geq \lceil n \log \left|\{x \in \mathcal{X} : P_X(x) > 0\}\right| \rceil. \tag{2.29}$$

**Example.** Consider $\mathcal{X} = [5]$ and a DMS with $P_X(x) = \frac{1}{5}$ for any $x \in \mathcal{X}$. To encode any symbol $X_i$ losslessly we need $\lceil \log 5 \rceil = 3$ bits. To encode a pair of symbols we need $\lceil \log 5^2 \rceil = 5$ bits, or 2.5 bits per symbol. To encode a triple we need $\lceil \log 5^3 \rceil = 7$ bits, or 2.33 bits per symbol. The entropy is $H(X) = \log 5 = 2.32$. We see that block-encoding is beneficial; in fact, the triple encoding is in this case very close to optimal already. For general (not uniform) sources, to achieve the entropy, we need to allow for a vanishing error.

Argue why this is the correct number of bits needed to losslessly compress the sting.

Comparing this to the bound $L \geq n\lceil \log \left|\{x \in \mathcal{X} : P_X(x) > 0\}\right| \rceil$, which we would have arrived at by encoding each source symbol separately using a fixed length code, we see that there is some improvement. However, the improvement is at most $n$ bits. Can we do better if we relax the stringent condition in (2.27) to be such that

$$P(\hat{X}^n \neq X^n) \leq \epsilon \tag{2.30}$$

for any $\epsilon > 0$ positive but arbitrarily small? Let us formalise this.

**Definition 2.7** (Block code). *An $(n, 2^L)$-fixed-length source code (or simply an $(n, 2^L)$-code) consists of an* encoder, *$e$, and a* decoder, *$d$, where*

- $e : \mathcal{X}^n \to \{0,1\}^L$ *and*
- $d : \{0,1\}^L \to \mathcal{X}^n$

The number $n$ is called the *block length* of the code; $L$ is the length of the codeword; and $R = \frac{L}{n}$ is called the rate of the code. The rate simply evaluates how many bits of codeword this codes uses per symbol of the source.

**Definition 2.8** (Achievable rate). *A rate $R$ is achievable for a DMS $\boldsymbol{X}$ if there exists a sequence of $(n, 2^{L_n})$-codes for $n \in \mathbb{N}$ and $L_n \in \mathbb{N}$ with encoder $e_n$ and decoder $d_n$ such that*

$$\limsup_{n\to\infty} \frac{L_n}{n} \leq R \qquad \text{and} \qquad \lim_{n\to\infty} P(\hat{X}^n \neq X^n) = 0 \tag{2.31}$$

*where*

$$\hat{X}^n = d_n(M), \quad \text{and} \quad M = e_n(X^n) \tag{2.32}$$

Using this definition, show that if $R$ is achievable so is any $R' \geq R$.

*are the reconstructed source and the codeword, respectively.*

Since we want to use as little memory as possible, what we really are interested in is the smallest $R$ that is still achievable.

There is a technicality in the definition of the achievable rate: since some limits might not exist we need to write lim sup instead of lim. This will however not effect any of our arguments.

**Definition 2.9** (Optimal source coding rate)**.** *The optimal source coding rate for the DMS $\boldsymbol{X}$, denoted as $R^*(\boldsymbol{X})$, is defined to be the infimum of all achievable rates, i.e.,*

$$R^*(\boldsymbol{X}) = \inf\{R : R \text{ is achievable for } \boldsymbol{X}\}. \tag{2.33}$$

Finding the optimal source coding rate looks like a formidable problem to solve at first sight. Note that the notion of achievable rate is asymptotic, namely we need to consider a sequence of codes, a code for each $n \in \mathbb{N}$, so that it is not even obvious that $R^*(\boldsymbol{X})$ is computable in finite time from a complexity-theoretic perspective. However, in his seminal work Shannon[3] showed that $R^*(X)$ has a simple form for a DMS.

[3] C. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 1948. DOI: 10.1002/j.1538-7305.1948.tb00917.x

**Theorem 2.10** (Fixed-length data compression)**.** *For any DMS $\boldsymbol{X}$ with pmf $P_X$, we have*

$$R^*(\boldsymbol{X}) = H(X) \tag{2.34}$$

To prove that $R^*(\boldsymbol{X}) = H(X)$, we must show the *achievability part*, $R^*(\boldsymbol{X}) \leq H(X)$, and the *converse part*, $R^*(\boldsymbol{X}) \geq H(X)$.

- Achievability means that, for every $R > H(X)$, we must exhibit a sequence of $(n, 2^{\lfloor nR \rfloor})$-codes such that the error probability vanishes as $n \to \infty$.
- The converse implies that we cannot do better than this, i.e., there is no sequence of $(n, 2^{\lfloor nR \rfloor})$-codes where $R < H(X)$ such that we have a vanishing error probability.

In most problems in information theory the two proofs (achievability and converse) use quite different techniques and we thus treat them separately.

### 2.3.2 *Proof of converse and Fano's inequality*

For the converse we will use Fano's inequality, which is a fundamental tool in the analysis of information processing tasks. We formulate it here as a statement about conditional entropies of strongly correlated random variables.

**Lemma 2.11** (Fano's inequality). *Let $X, Y$ be random variables with joint pmf $P_{XY}$ and let $\epsilon := P[X \neq Y]$. Then*

$$H(X|Y)_P \leq h(\epsilon) + \epsilon \log(|X| - 1) \leq 1 + \epsilon \log |X| \,, \tag{2.35}$$

*where $h(\epsilon) = -\epsilon \log \epsilon - (1-\epsilon)\log(1-\epsilon)$ is the binary entropy.*

This essentially tells us that if the probability that the two random variables differ is small, then so is the conditional entropy. We should expect this, since the conditional entropy measures the uncertainty of $X$ given side information $Y$: if $\epsilon$ is small then knowing the value $y$ of $Y$ allows us to guess that $X = y$ as well, which is correct with high probability.

*Proof.* First, we recall that $H(X|Y)_P = \sum_y P_Y(y) H(X|Y=y)$. We will bound these terms individually first. Define $\epsilon_y = 1 - P_{X=y|Y=y}$ so that $\sum_y P_Y(y)\epsilon_y = \epsilon$. Then, we find

$$\begin{aligned}
H(X|Y=y) &= -\sum_x P_{X|Y=y}(x) \log P_{X|Y=y}(x) && (2.36)\\
&= -(1-\epsilon_y)\log(1-\epsilon_y) - \sum_{x\neq y} P_{X|Y=y}(x) \log P_{X|Y=y}(x) && (2.37)\\
&= -(1-\epsilon_y)\log(1-\epsilon_y) - \epsilon_y \log \epsilon_y \\
&\qquad - \epsilon_y \sum_{x\neq y} \frac{P_{X|Y=y}(x)}{\epsilon_y} \log \frac{P_{X|Y=y}(x)}{\epsilon_y} && (2.38)\\
&= h(\epsilon_y) - \epsilon_y \sum_{x\neq y} \frac{P_{X|Y=y}(x)}{\epsilon_y} \log \frac{P_{X|Y=y}(x)}{\epsilon_y} \,. && (2.39)
\end{aligned}$$

To get from (2.37) to (2.38) we used that

$$\begin{aligned}
&\epsilon_y \sum_{x\neq y} \frac{P_{X|Y=y}(x)}{\epsilon_y} \log \frac{P_{X|Y=y}(x)}{\epsilon_y} \\
&\quad = \sum_{x\neq y} P_{X|Y=y}(x) \log P_{X|Y=y}(x) - \sum_{x\neq y} P_{X|Y=y}(x) \log \epsilon_y && (2.40)\\
&\quad = \sum_{x\neq y} P_{X|Y=y}(x) \log P_{X|Y=y}(x) - \epsilon_y \log \epsilon_y \,. && (2.41)
\end{aligned}$$

Finally, we observe that the second term in (2.39) is $\epsilon_y$ times the entropy of the pmf $\frac{1}{\epsilon_y} P_{X|Y=y}$ supported on all $x \neq y$, and the entropy is upper bounded by the logarithm of the support as shown in Chapter 1. This yields

$$H(X|Y=y) \leq h(\epsilon_y) + \epsilon_y \log(|X| - 1) \tag{2.42}$$

It remains to take an average of the above bound. Using (once again) concavity of the entropy, we find

$$\sum_y P_Y(y) H(X|Y=y) \leq \sum_y P_Y(y) h(\epsilon_y) + \epsilon_y \log(|X|-1) \tag{2.43}$$

$$\leq h\Big(\sum_y P_Y(y)\epsilon_y\Big) + \epsilon \log(|X|-1) \tag{2.44}$$

$$= h(\epsilon) + \epsilon \log(|X|-1)\,. \tag{2.45}$$

Finally, we can bound $h(\epsilon) \leq \log 2 = 1$ and $|X|-1 \leq |X|$ to make the bound a bit simpler but still sufficiently tight for most purposes. □

*Proof of converse of Theorem 2.10.* Consider any sequence of $(n, 2^{L_n})$-codes with encoders $e_n$ and decoders $d_n$ that satisfy $\lim_{n\to\infty} \epsilon_n = 0$, where the decoding error is

$$\epsilon_n := P(\hat{X}^n \neq X^n) \tag{2.46}$$

with $\hat{X}^n = d_n(e_n(X^n))$. Fano's inequality applied to the estimation of $X^n$ yields

$$H(X^n|\hat{X}^n) \leq \epsilon_n n \log|\mathcal{X}| + 1 \tag{2.47}$$

Since $H(X^n|M) \leq H(X^n|\hat{X}^n)$ by the data-processing inequality for the conditional entropy (see Corollary 1.10) applied to the decoder $d_n$, this can be relaxed to

$$H(X^n|M) \leq \epsilon_n n \log|\mathcal{X}| + 1 \tag{2.48}$$

Furthermore, using the dimension bound for $|M| = 2^{L_n}$, and the definition of mutual information, we find

$$L_n \geq H(M) \tag{2.49}$$

$$\geq I(X^n : M) \tag{2.50}$$

$$= nH(X) - H(X^n|M) \tag{2.51}$$

We can now apply Eq. (2.48) to get

$$\frac{L_n}{n} \geq H(X) - \epsilon_n \log|\mathcal{X}| - \frac{1}{n}\,. \tag{2.52}$$

Now we can take the limit $n \to \infty$ on both sides, which yields

$$\limsup_{n\to\infty} \frac{L_n}{n} \geq H(X) \tag{2.53}$$

since $\epsilon_n \to 0$ as $n \to \infty$. Hence, there cannot exist a sequence of codes with vanishing error that satisfies $\limsup_{n\to\infty} \frac{L_n}{n} \leq R$ for any $R < H(X)$, which is what we set out to show. Hence, $R^*(\boldsymbol{X}) \geq H(X)$. □

### 2.3.3 *Proof of achievability and typical sets*

The main idea in the proof of achievability is to only encode sequences of source outputs that are "typical" in a sense we will make precise below. The sets of typical sequences are chosen in such a way that two crucial properties hold:

- There are not too many such sequences so that we can encode them using not too many bits.
- We can safely ignore all the sequences that are not typical since they are guaranteed to only occur with very low probability.

Let us now make this more formal. We start by defining the typical set and will then show its properties.

Let $\mu \in (0,1)$ and consider a DMS $\boldsymbol{X}$. The $\mu$-typical set for $\boldsymbol{X}$, for each $n \in \mathbb{N}$, is defined as

$$A_\mu^{(n)}(\boldsymbol{X}) := \left\{ x^n \in \mathcal{X}^n : \left| \frac{1}{n} \log \frac{1}{P_{X^n}(x^n)} - H(X) \right| \leq \mu \right\} \tag{2.54}$$

where

$$P_{X^n}(x^n) = P(X^n = x^n) = \prod_{i=1}^{n} P_X(x_i), \qquad \forall\, x^n \in \mathcal{X}^n. \tag{2.55}$$

We call the elements of $A_\mu^{(n)}(\boldsymbol{X})$ *$\mu$-typical sequences* of length $n$ for the source $\boldsymbol{X}$. In words, this means that the typical sequences are those whose average surprisal is very close to the entropy of $X$, the i.i.d. output of the source. Note that

$$H(X) = \frac{1}{n} H(X_1 X_2 \dots X_n) \tag{2.56}$$

due to the additivity of entropy for product distributions, and thus we can alternatively interpret $H(X)$ as the entropy the source creates per symbol.

The properties of the typical set mentioned above can now be formalised as the asymptotic equipartition property (AEP).

**Proposition 2.12** (AEP)**.** *Let $\mu \in (0,1)$. The sequence of typical sets $A_\mu^{(n)}(\boldsymbol{X})$ for $n \in \mathbb{N}$ has the following properties:*

1. $H(X) - \mu \leq \frac{1}{n} \log \frac{1}{P_{X^n}(x^n)} \leq H(X) + \mu$ *for all sequences* $x^n \in A_\mu^{(n)}(\boldsymbol{X})$ *and* $n \in \mathbb{N}$.
2. $\lim_{n\to\infty} P\big[X^n \in A_\mu^{(n)}(\boldsymbol{X})\big] = 1$.

3. *For all $n \in \mathbb{N}$, the size of the set satisfies*

$$\left|A_\mu^{(n)}(X)\right| \leq 2^{n(H(X)+\mu)}\,. \tag{2.57}$$

The name *asymptotic equipartition property* alludes to the the first (and defining) property of the typical set, which ensures that all sequences in the set are approximately equally likely. More precisely, the definition implies that we have

$$\left|\frac{P_{X^n}(x^n)}{P_{X^n}(\tilde{x}^n)}\right| \leq \exp(2n\mu) \tag{2.58}$$

for all typical sequences $x^n, \tilde{x}^n \in A_\mu^{(n)}(\boldsymbol{X})$.

*Proof.* The first property is immediate from the definition of $A_\mu^{(n)}(X)$.

The second property follows from the weak law of large numbers. To see this, we consider the random variables $X_i$ produced by the source and the new random variables

Verify that $Z_i$ has zero mean and that the sequence $(Z_1, Z_2, \ldots, Z_n)$ is i.i.d..

$$Z_i = \log \frac{1}{P_X(X_i)} - H(X). \tag{2.59}$$

We would now like to express the probability $P\big[X^n \in A_\mu^{(n)}(\boldsymbol{X})\big]$ in terms of the random variables $Z_i$ we just introduced. We find the following sequence of equalities:

$$\begin{aligned}
P\big[X^n \in A_\mu^{(n)}(\boldsymbol{X})\big] &= P\left[\left|\frac{1}{n}\log\frac{1}{P_{X^n}(X^n)} - H(X)\right| \leq \mu\right] && (2.60)\\
&= P\left[\left|\frac{1}{n}\sum_{i=1}^n\left(\log\frac{1}{P_X(X_i)} - H(X)\right)\right| \leq \mu\right] && (2.61)\\
&= P\left[\left|\frac{1}{n}\sum_{i=1}^n Z_i\right| \leq \mu\right] && (2.62)\\
&= 1 - P\left[\left|\frac{1}{n}\sum_{i=1}^n Z_i\right| > \mu\right]. && (2.63)
\end{aligned}$$

Now since $Z_i$ are i.i.d. and zero mean we can apply the weak law of large numbers (Proposition 0.5), which ensures that

$$\lim_{n\to\infty} P\left[\left|\frac{1}{n}\sum_{i=1}^n Z_i\right| > \mu\right] = 0\,, \tag{2.64}$$

and so, in particular, we have $\lim_{n\to\infty} P\big[X^n \in A_\mu^{(n)}(\boldsymbol{X})\big] = 1$.

The third property follows by a basic counting argument. Since every sequence in $A_\mu^{(n)}(\boldsymbol{X})$ has probability at least $2^{-n(H(X)+\mu)}$ by definition, there can be at most $2^{n(H(X)+\mu)}$ such sequences in the typical set as otherwise the total probability of all sequences would exceed 1. □

*Proof of achievability of Theorem 2.10.* We fix $\mu \in (0,1)$ for the moment and construct encoders and decoders for each block length $n$, targeting a rate $R = H(X) + \mu$. The main idea is to do a faithful encoding of all the sequences $x^n$ in the $\mu$-typical set and essentially ignore sequences that are not typical.

To do this we index the elements of $A_\mu^{(n)}(\boldsymbol{X})$ in some order (say lexicographic). This simply means that to each sequence $x^n \in A_\mu^{(n)}(X)$, we assign a unique index $\operatorname{idx}(x^n) \in \{0,1\}^{L_n}$, where $L_n$ is chosen large enough so that all typical sequences fit in. This assignment function and its inverse are known to both the encoder and the decoder. We first give a bound on the length $L_n$ required to store typical sequences. We find $L_n = \lceil \log |A_\mu^{(n)}(\boldsymbol{X})| \rceil \leq \lceil n(H(X)+\mu) \rceil \leq nH(X) + n\mu + 1$ suffices, which implies that $\lim_{n\to\infty} \frac{L_n}{n} \leq H(X) + \mu$.

We can now design the encoder and decoder for block length $n$.

*Encoder $e_n$:* If the realised sequence is typical, i.e. $x^n \in A_\mu^{(n)}(\boldsymbol{X})$, then output the index $M = \operatorname{idx}(X^n)$. Otherwise set $M$ to any value in the image of the function idx, say $0^{L^n}$. In other words, the precise working of the encoder is

$$e_n(x^n) = \begin{cases} \operatorname{idx}(x^n) & x^n \in A_\mu^{(n)}(\boldsymbol{X}) \\ 0^{L_n} & x^n \notin A_\mu^{(n)}(\boldsymbol{X}) \end{cases} . \tag{2.65}$$

*Decoder $d_n$:* Output $x^n \in A_\mu^{(n)}(\boldsymbol{X})$ such that $m = \operatorname{idx}(x^n)$, or, in other words, output $d_n(m) = \operatorname{idx}^{-1}(m)$.

Next we compute the probability of error for this encoder and decoder, which we denote by $\epsilon_n$ as usual. Suppose first that the realised sequence is typical. We will never make an error because the sequence that is output coincides with the emitted sequence of the DMS, by construction of the encoder and decoder. Thus, we can only make an error if the emitted source sequence is atypical. Formally, we can bound

$$\begin{aligned} \epsilon_n &= P\big[X^n \neq \hat{X}^n\big] && (2.66)\\ &= P\big[X^n \neq \hat{X}^n \big| X^n \in A_\mu^{(n)}(X)\big] P[X^n \in A_\mu^{(n)}(X)] \\ &\quad + P\big[X^n \neq \hat{X}^n \big| X^n \notin A_\mu^{(n)}(X)\big] P[X^n \notin A_\mu^{(n)}(X)] && (2.67)\\ &\leq P\big[X^n \notin A_\gamma^{(n)}(X)\big], && (2.68) \end{aligned}$$

where the last inequality follows from the fact that

$$P\big[X^n \neq \hat{X}^n \big| X^n \in A_\mu^{(n)}(X)\big] = 0 \tag{2.69}$$

by design of our encoder and decoder and the simple observation that $P\big[X^n \neq \hat{X}^n \big| X^n \notin A_\mu^{(n)}(X)\big] \leq 1$. Thus, we can conclude that $\lim_{n\to\infty} \epsilon_n = 0$ by the property of the typical set.

This code thus exhibits the fact that the rate $H(X) + \mu$ is achievable. Finally, since $\mu > 0$ is arbitrarily small, we have

$$R^*(\boldsymbol{X}) = \inf\{R : R \text{ is achievable on } \boldsymbol{X}\} \tag{2.70}$$
$$\leq \inf_{\mu>0}\{H(X) + \mu\} = H(X). \tag{2.71}$$

□

### 2.3.4 *Strong converse via typical sets*

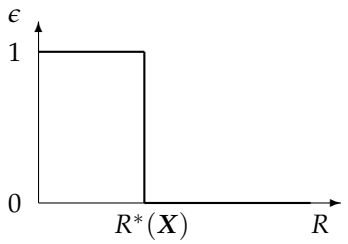



Figure 2.5: **Illustration of the strong converse property.** For any rate sequence of $(n, 2^{nR})$ codes with rate $R < R^*(\boldsymbol{X}) = H(X)$, the asymptotic error $\lim_{n\to\infty} P(\hat{X}^n \neq X^n)$ necessarily converges to 1.

We can also argue with typical sets to get a stronger statement for our converse bound. In fact, the converse via Fano's inequality is quite conservative. Even if we allow that

$$\limsup_{n\to\infty} P(\hat{X}^n \neq X^n) \leq \epsilon \tag{2.72}$$

for any $\epsilon \in [0, 1)$, it turns out that the $\epsilon$-optimal rate must be no smaller than $H(X)$. We state this formally as follows:

**Theorem 2.13.** *For any sequence of $(n, 2^{L_n})$-codes with*

$$\limsup_{n\to\infty} \frac{L_n}{n} < H(X), \tag{2.73}$$

*it necessarily holds that $P(\hat{X}^n \neq X^n) \to 1$ as $n \to \infty$.*

This theorem removes all hope to devise a more efficient source coding scheme that can beat the compression rate $H(X)$ by allowing for some small (or even large) error.

One way to prove this statement is to expand on our characterisation of the typical set. In particular, we want to show that the typical set cannot be too small.

**Proposition 2.14** (AEP, continued)**.** *Let $\nu, \mu \in (0, 1)$. Then there exists an $N_0$ such that for all $n \geq N_0$, the following holds:*

1. $P\big[X^n \in A_\mu^{(n)}(\boldsymbol{X})\big] \geq 1 - \nu$.

2. *The size of the set satisfies*

$$\left|A_\mu^{(n)}(X)\right| \geq (1-\nu)2^{n(H(X)-\mu)}\,. \tag{2.74}$$

*Proof.* The first statement is a direct consequence of the second property in Proposition 2.12, i.e., since $\lim_{n\to\infty} P\big[X^n \in A_\mu^{(n)}(\boldsymbol{X})\big] = 1$ there must exist a constant $N_0 \in \mathbb{N}$ with the desired property by definition of the limit.

The second statement again follows by a counting argument. Since we know that every sequence $x^n \in A_\mu^{(n)}(X)$ satisfies $P_{X^n}(x^n) \leq 2^{-n(H(X)-\mu))}$ we will need at least $(1-\nu)2^{n(H(X)-\mu)}$ such elements to reach a total probability of $1-\nu$ as stipulated by the first property above. This yields the lower bound on the cardinality of the set. □

This allows us to lay out the idea for a proof of Theorem 2.13. Let us assume that we target a rate $R = \limsup_{n\to\infty} \frac{L_n}{n} < H(X)$ and define $\mu = \frac{1}{3}\big(H(X) - R\big)$. First, we can observe that $L_n \leq n(R+\mu)$ for large enough $n$ by the definition of the lim sup. Then using the bound in Eq. (2.74), we find

$$2^{L_n} \leq 2^{n(R+\mu)} = 2^{n(H(X)-2\mu)} \leq \frac{2^{-n\mu}}{1-\nu}\left|A_\mu^{(n)}(\boldsymbol{X})\right|\,. \tag{2.75}$$

For sufficiently large $n$ this implies that $2^{L_n} < |A_\mu^{(n)}(\boldsymbol{X})|$, and in fact the set $\{0,1\}^{L_n}$ is smaller by a factor that grows exponentially in $n$. This implies that we can only faithfully represent a smaller and smaller fraction of all typical source sequences in $\{0,1\}^{L_n}$. Moreover, since all typical sequences are almost equiprobable this induces an error approaching 1 exponentially fast. With this we can now give a proof of the strong converse.

*Proof of Theorem 2.13.* We assume $R = \limsup_{n\to\infty} \frac{L_n}{n} < H(X)$ and aim to give a lower bound on the probability of error, in fact, we aim to show that the error must tend to 1.

Let us first fix $\mu \in (0,1)$ such that $R \leq H(X) - 3\mu$. This implies that, for large enough $n$, we must have

$$\frac{L_n}{n} < H(X) - 2\mu \tag{2.76}$$

by the definition of the lim sup.

We also fix $\nu \in (0,1)$ for now, although it is important that we can choose it arbitrarily small. According to Proposition 2.14, we can always find an $N_0$ such that $P\big[X^n \notin A_\mu^{(n)}(\boldsymbol{X})\big] \leq \nu$ for all $n \geq N_0$.

For such $n$, let us now consider an arbitrary encoder $e_n : \mathcal{X}^n \to \{1,2,\ldots,2^{L_n}\}$. This encoder induces a partition of the space $\mathcal{X}^n$ into disjoint subsets $\mathcal{D}_m \subset \mathcal{X}^n, m \in \{1,2,\ldots,2^{L_n}\}$ defined as $\mathcal{D}_m = \{x^n :$

$e_n(x^n) = m\}$. The best decoder (the one that minimises the error probability) is the one that uses the following rule given message $m$:

$$d_n(m) = \operatorname*{argmax}_{x^n \in \mathcal{D}_m} P_{X^n}(x^n). \tag{2.77}$$

This is known as the maximum likelihood decoder as the decoder maximizes the likelihood of the observed data. We also denote the resulting random variable by $\hat{X}^n = d_n(e_n(X^n))$ as usual.

The error probability can be bounded as

$$\begin{aligned}
\epsilon_n &= 1 - P(\hat{X}^n = X^n) && (2.78)\\
&= 1 - \sum_m \sum_{x^n \in \mathcal{D}_m} P_{X^n}(x^n) P(\hat{X}^n = X^n | X^n = x^n) && (2.79)\\
&= 1 - \sum_m P_{X^n}(d_n(m)) && (2.80)\\
&\geq 1 - \nu - \sum_m \mathbf{1}\{d_n(m) \in \mathcal{A}_\mu^{(n)}(\boldsymbol{X})\}\, P_{X^n}(d_n(m))\,, && (2.81)
\end{aligned}$$

where the last inequality follows because the probability of the atypical set is smaller than $\nu$. Since the probability of sequences in the typical set is at most $2^{-n(H(X)-\mu)}$, we can now further bound

$$\begin{aligned}
\epsilon_n &\geq 1 - \nu - \sum_m \mathbf{1}\{d_n(m) \in \mathcal{A}_\mu^{(n)}(\boldsymbol{X})\}\, 2^{-n(H(X)-\mu)} && (2.82)\\
&\geq 1 - \nu - \sum_m 2^{-n(H(X)-\mu)} && (2.83)\\
&= 1 - 2^{L_n - n(H(X)-\mu)} - \nu && (2.84)
\end{aligned}$$

Hence, for large enough $n$ we can use (2.76) to deduce that

$$\epsilon_n \geq 1 - \nu - 2^{-n\mu}\,; \tag{2.85}$$

but this implies that $\liminf_{n\to\infty} \epsilon_n \geq 1 - \nu - \lim_{n\to\infty} 2^{-n\mu} = 1 - \nu$. Since $\nu$ can be arbitrarily small, we in fact must have $\lim_{n\to\infty} \epsilon_n = 1$, which is what we aimed to show. □

## 2.4 *Exercises*

**Exercise 2.1** (Kraft–McMillan inequality)**.** *Assume a uniquely decodable code has codeword lengths $l_1, \ldots, l_M$. Our goal is to derive Kraft's inequality for uniquely decodable codes:*

$$\sum_{j=1}^{M} 2^{-l_j} \leq 1\,.$$

a) *Prove the following identity (this is easy):*

$$\left(\sum_{j=1}^{M} 2^{-l_j}\right)^n = \sum_{j_1=1}^{M} \sum_{j_2=1}^{M} \cdots \sum_{j_n=1}^{M} 2^{-(l_{j_1} + l_{j_2} + \ldots + l_{j_n})}.$$

b) *Let $A_l$ be the number of concatenations of $n$ codewords that have overall length $l = l_{j_1} + l_{j_2} + \ldots + l_{j_n}$ and let $l_{\max} = \max\{l_1, l_2, \ldots, l_M\}$ be the maximum length of a codeword. Show that*

$$\left(\sum_{j=1}^{M} 2^{-l_j}\right)^n = \sum_{l=n}^{nl_{\max}} A_l \, 2^{-l}.$$

c) *Using unique decodability, show that $A_i \leq 2^i$ and hence*

$$\left(\sum_{j=1}^{M} 2^{-l_j}\right)^n \leq nl_{\max}.$$

*Use this to derive the desired inequality.*

**Exercise 2.2.** *Let $X$ be an i.i.d. random variable with an infinite alphabet, $\mathcal{X} = \{1, 2, 3, \ldots\}$. In addition, let $P(X = i) = 2^{-i}$.*

a) *What is the entropy of $X$?*

b) *Find an optimal variable length code, and show that it is indeed optimal.*

**Exercise 2.3** (Shannon Code)**.** *Let $X$ be a random variable distributed on $\{0, 1, 2, \ldots, d-1\}$, and let $P_X$ be its pmf. Without loss of generality, we assume $P_X(0) \geq P_X(1) \geq \cdots \geq P_X(d-1) > 0$.*

*A Shannon code for this source is constructed as follows. For each $x \in \mathcal{X}$, we consider the binary representation of the real number $\sum_{x'<x} P_X(x')$, and use the first $\lceil \log \frac{1}{P_X(x)} \rceil$ bits in its fractional part to represent $x$. E.g., the binary representation of $1/3$ is '0.01010101…', and thus we assign it codeword '01'.*

a) *Show that the above code is unique decodable.*

b) *Produce a Shannon code table for the following source*

| $x$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| $P_X(x)$ | 1/2 | 1/6 | 1/6 | 1/6 |

*and compute the expected code length.*

c) *Verify that the expected length of Huffman codes for this source is shorter.*

**Exercise 2.4.** *Which of the following sets of codewords can never be valid Huffman codes (for any source probabilities)? Argue why.*

a) $\{00, 01, 10, 110\}$,

b) $\{0, 10, 11\}$,

c) $\{00, 01, 10, 11\}$,

d) $\{01, 10\}$,

e) $\{1, 01, 10\}$.

**Exercise 2.5.** *Consider a random variable $X$ which takes on four possible values with probabilities $(\frac{1}{3}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12})$.*

a) *Construct a Huffman code for this source.*

b) *Show that there exist two different sets of optimal lengths for the codewords, namely, the codeword length assignments $(1, 2, 3, 3)$ and $(2, 2, 2, 2)$ are both optimal.*

c) *Are there optimal codes with codeword lengths for some symbols that exceed the Shannon code length $\lceil \log \frac{1}{p(x)} \rceil$?*

**Exercise 2.6.** *Suppose there is a source $X$ with alphabet $\{1, 2, \ldots, n\}$ and each symbol has equal probability. We use the Huffman algorithm to generate a binary code. Calculate the average code length when $n = 2^L$ and $n = 2^L + 1$, where $L$ is a positive integer.*

**Exercise 2.7** (Huffman algorithm)**.** *Consider a discrete memoryless source $X$ with alphabet $\{1, 2, \ldots, M\}$. Suppose that the symbol probabilities are ordered and satisfy $p_1 > p_2 > \ldots > p_M$ and also satisfy $p_1 < p_{M-1} + p_M$. Let $l_1, l_2, \ldots, l_M$ be the lengths of a prefix-free code of minimum expected length for such a source.*

a) *Is the following statement true or false? $l_1 \leq l_2 \leq \ldots \leq l_M$. Argue why.*

b) *Show that if the Huffman algorithm is used to generate the above code, then $l_M \leq l_1 + 1$.*

**Hint:** A minimum-expected-length code must be full.

c) *Show that $l_M \leq l_1 + 1$ for* any *(not necessarily Huffman generated) prefix-free code of minimum expected length.*

d) *Suppose $M = 2^k$ for some integer $k$. Show that all codewords must have the same length.*

**Hint:** What does the Kraft inequality look like for an optimal code? Consider the three cases $l_1 = k$, $l_1 < k$ and $l_1 > k$.

**Exercise 2.8** (Compressing English)**.** *Consider a source that outputs independent letters according to the frequency that they appear in the English language (use the frequencies listed in Table 2.1).*

a) *Calculate the entropy of this source.*

b) *Construct a Huffman code for this source. You may either construct the Huffman code manually according to the algorithm discussed in the lecture, or write a compute program that does this for you.*

c) *Compute the expected length of the codeword for this code. How does this compare to the entropy computed in a)?*

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| a | 8.4% | b | 1.5% | c | 2.2% | d | 4.2% | e | 11.0% | f | 2.2% |
| g | 2.0% | h | 6.0% | i | 7.4% | j | 0.1% | k | 1.3% | l | 4.0% |
| m | 2.4% | n | 6.7% | o | 7.4% | p | 1.9% | q | 0.1% | r | 7.5% |
| s | 6.2% | t | 9.2% | u | 2.7% | v | 0.9% | w | 2.5% | x | 0.1% |
| y | 2.0% | z | 0.1% | | | | | | | | |

Table 2.1: Statistical distribution of letters in the English language. Source: https://en.wikipedia.org/wiki/Letter_frequency, but normalized so that they add up to 100%.

**Exercise 2.9** (Code with Prefix)**.** *We usually like codes to be prefix-free as otherwise we do not know when a codeword ends and decoding might no longer be unique. A simple alternative way to overcome this problem is to add a special symbol that indicates the end of a codeword. In this exercise we thus use codewords comprised of "0" and "1" that always end with "_".*

a) *Construct a code for this source in the following way: the two most frequent letters according to Table 2.1 are assigned codewords of length* $1+1$*, the next four most frequent letters codewords of lenght* $2+1$*, etc.*

b) *Compute the expected length of the codewords.*

c) *The code you arrived at is very similar to a famous code that has been in use since the 1830s. Which code is that? Compare the expected length of codewords of your code to that of this historical code.*

**Hint:** Replace "0" by "·" and "1" by "−" in your codewords.

**Exercise 2.10.** *Consider a DMS with a two symbol alphabet* $\{a, b\}$ *where* $p_X(a) = 2/3$ *and* $p_X(b) = 1/3$*. Let* $X^n = (X_1, \ldots, X_n)$ *be a string of symbols emitted by the source with* $n = 100,000$*. Let* $W(X_j)$ *be the suprisal for the j-th source output, i.e.,* $W(X_j) = -\log \frac{2}{3}$ *for* $X_j = a$ *and* $-\log \frac{1}{3}$ *for* $X_j = b$*. Define* $W(X^n) = \sum_{j=1}^{n} W(X_j)$*.*

a) *Find the variance of* $W(X_j)$*. For* $\epsilon = 0.01$*, evaluate a bound on the probability of the typical set* $\mathcal{A}_\epsilon^{(n)}$ *using Chebyshev's inequality.*

b) *Let* $N_a$ *be the number of a's in the string* $X^n = (X_1, \ldots, X_n)$ *. The rv* $N_a$ *is the sum of n i.i.d. rv's. What are these?*

c) *Express the rv* $W(X^n)$ *as a function of* $N_a$*. Note how this depends on n.*

d) *Express the typical set in terms of bounds on* $N_a$*. Use Chebyshev's inequality to derive bounds on the probability of the typical set, using properties of* $N_a$ *instead of* $W(X_j)$*.*

**Hint:** You may write $\mathcal{A}_\epsilon^{(n)} = \{x^n : \alpha < N_a < \beta\}$ and calculate $\alpha$ and $\beta$.

e) *Find* $P[N_a = i]$ *for* $i = 0, 1, 2$*. Find the probability of each individual string* $x^n$ *for those values of i. Find the particular string* $x^n$ *that has maximum probability over all sample values of* $X^n$*. What are the next most probable n-strings? Give a brief discussion of why the most probable n-strings are not regarded as typical strings.*

**Exercise 2.11.** *Consider a DMS* **X** *which produces sequences of letters from an alphabet* $\mathcal{X}$ *such that the entropy of each letter is* $H(X)$*. Assume also*

*that for any $k \in \mathbb{N}$, we have a Huffman code for k-tuples produced by this source. That is, a Huffman code that maps sequences $x^k \in \mathcal{X}^k$ of length k to codewords $C_k(x^k) \in \{0,1\}^*$ with lengths $\ell_k(x^k)$.*

a) *Give an upper bound on $\mathbb{E}[\ell_k(X^k)]$ for $X^k$ produced from this source in terms of the entropy $H(X)$. You might want to consider the case $k = 1$ first and recall properties of the Shannon code and its relation to the Huffman code.*

b) *Use the weak law of large numbers to show that, for any $k \in \mathbb{N}$,*

$$\lim_{m\to\infty} P\left[\frac{1}{mk}\sum_{i=1}^{m} \ell_k(X_i^k) \geq H(X) + \frac{2}{k}\right] = 0\,.$$

c) *Now consider a rate $R = H(X) + \delta$ for any $\delta > 0$. Choose $k = \lceil \frac{2}{\delta} \rceil + 1$. For each $m \in \mathbb{N}$, design a block encoder for blocks of length $n = km$, that is, an encoder $e_n : X^n \to \left[2^{\lfloor Rn \rfloor}\right]$ and a decoder $d_n$ creating an estimate $\hat{X}^n$ such that*

$$\lim_{n\to\infty} P\left[X^n \neq \hat{X}^n\right] = 0\,.$$

*You have now shown that optimal variable-length codes are also optimal also for block-coding.*

**Exercise 2.12** (Coding with side information)**.** *Consider a memoryless source $(\mathbf{X}, \mathbf{Y})$ that produces in each iteration two random variables, $X_i$ and $Y_i$, where $X_i$ is private information and $Y_i$ is public information. The pairs $(X_i, Y_i)$ are i.i.d. following a pmf $P_{XY}$.*

*We are looking for a fixed-length block code that compresses the private information $X^n = (X_1, X_2, \ldots, X_n)$ using the public information $Y^n = (Y_1, Y_2, \ldots, Y_n)$ such that the code can be decoded asymptotically error-free with help of the public information.*

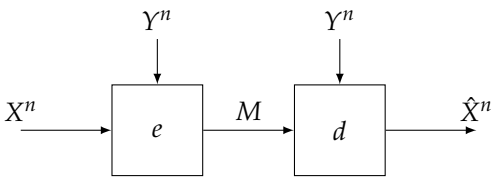



Figure 2.6: An $(n, 2^L)$-code for a source with side information is given by an encoder, $e : (X^n, Y^n) \to M$, and decoder, $d : (M, Y^n) \to \hat{X}^n$. The codeword $M \in \{0,1\}^L$ is of length $L$.

*We define $R^*(\mathbf{X}|\mathbf{Y})$ as the infimum over all rates R such that there exists a sequence of $(n, 2^{nR})$-codes satisfying*

$$\lim_{n\to\infty} P\left[X^n \neq \hat{X}^n\right] = 0, \quad \textit{where} \quad \hat{X}^n = d_n(e_n(X^n, Y^n), Y^n)\,.$$

a) *Determine $R^*(\mathbf{X}|\mathbf{Y})$, by intuitive or formal arguments, for the simple cases where*

*(a) $X$ and $Y$ are independent,*

*(b) $X = Y$;*

b) *By explicitly constructing a code for the source $(X, Y)$ using codes for the sources $Y$ and $X$ (with side information $Y$), show that $R^*(\mathbf{X}, \mathbf{Y}) \leq R^*(\mathbf{X}|\mathbf{Y}) + R^*(\mathbf{Y})$.*

c) *Show the converse, $R^*(\mathbf{X}|\mathbf{Y}) \geq H(X|Y)$, using Fano's inequality.*

d) *Give a formal proof or a sketch of a proof that $R^*(\mathbf{X}|\mathbf{Y}) \leq H(X|Y)$. For this purpose, consider the conditional typical set $\mathcal{A}_\epsilon^{(n)}(\mathbf{X}|\mathbf{Y})$, given as*

$$\left\{(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \left|\frac{1}{n}\log\frac{1}{P_{X^n|Y^n}(x^n|y^n)} - H(X|Y)\right| \leq \epsilon\right\}.$$

*This establishes that $R^*(\mathbf{X}|\mathbf{Y}) = H(X|Y)$.*

You may use the following sequence of inequalities, which needs to be verified.

$$\begin{aligned} H(X^n|\hat{X}^n) &\geq H(X^n|Y^n M) \\ &= H(X^n M|Y^n) - H(M|Y^n) \\ &\geq H(X^n M|Y^n) - L \\ &\geq H(X^n|Y^n) - L\,. \end{aligned}$$

**Exercise 2.13** (Slepian–Wolf Coding). *Let $X$ and $Y$ be a pair of jointly distributed random variables on finite alphabets $\mathcal{X}$ and $\mathcal{Y}$, respectively.*

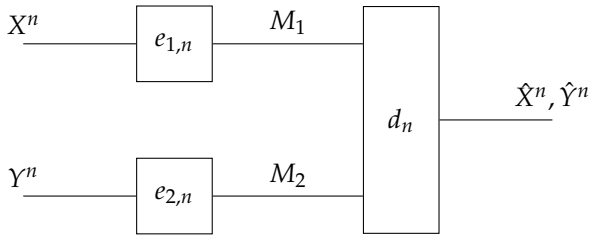



Figure 2.7: An $(n, 2^{nL_1}, 2^{nL_2})$-separately-encoded jointly-decoded source code consists of a pair of encoders $e_1 : \mathcal{X}^n \to \{0,1\}^{nL_1}$ and $e_2 : \mathcal{Y}^n \to \{0,1\}^{nL_2}$, and a decoder $d : \{0,1\}^{nL_1} \times \{0,1\}^{nL_2} \to \mathcal{X}^n \times \mathcal{Y}^n$.

*The rate pair $(R_1, R_2)$ is said to be achievable for DMS $(X, Y)$ if there exists a sequence of $(n, 2^{nL_1}, 2^{nL_2})$-codes with encoders $e_{1,n}$, $e_{2,n}$ and decoder $d_n$ such that*

$$\lim_{n\to\infty} P\{(\hat{X}^n, \hat{Y}^n) \neq (X^n, Y^n)\} = 0$$

*where*

$$(\hat{X}^n, \hat{Y}^n) = d_n(M_1, M_2),\ M_1 = e_{1,n}(X^n),\ \text{and}\ M_2 = e_{2,n}(Y^n)$$

*are the reconstructed source and codewords respectively.*

*Prove that, for any $(R_1, R_2)$ achievable, it must hold that*

$$\begin{aligned} R_1 &\geq H(X|Y), \\ R_2 &\geq H(Y|X), \qquad \text{and} \\ R_1 + R_2 &\geq H(X, Y). \end{aligned}$$

# 3
# *Statistics: Binary hypothesis testing*

**Intended learning outcomes:**

- You understand the setup of symmetric and asymmetric binary hypothesis testing.
- You can compute the minimal error probability in binary hypothesis testing with known priors, and understand its relationship with the total variation distance.
- You can determine the type of a sequence and its empirical distribution.
- You are able to apply and analyse threshold tests.
- You can apply the Chernoff exponent and Stein's lemma.

**Book reference:** Chapter 11, Sections 11,1 and 11.7–11.9 in Cover & Thomas[1], but we are not following it too closely.

[1] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991. ISBN 9780471748823. DOI: 10.1002/047174882X

## *3.1 Problem setup and definitions*

We consider binary hypothesis testing where we try to distinguish between two models of a random process. The random process produces a sequence of random variables $\boldsymbol{X} = (X_1, X_2, \ldots)$ that are independently drawn from some (unknown) probability distribution $Q \in \mathcal{P}(\mathcal{X})$, where we take $\mathcal{X}$ to be any discrete set. Consider the hypothesis test between two hypotheses, $H_0$ and $H_1$:

$$\begin{aligned} H_0 &: Q = P_0 \\ H_1 &: Q = P_1, \end{aligned} \tag{3.1}$$

where $P_0, P_1 \in \mathcal{P}(\mathcal{X})$ are two candidate probability distributions (or models) of the process creating the random outcome. Our goal is to deduce, from the observation of the random sequence $\boldsymbol{X}$, which of the two hypotheses is correct. $H_0$ is usually called the *null-hypothesis* and $H_1$ the *alternate hypothesis*.

A (deterministic) *test* for the sequence $X^{(n)} = (X_1, X_2, \ldots, X_n)$ is a region $\mathcal{A}_n \subset \mathcal{X}^n$. We say that the alternate hypothesis is *accepted* for this test if the observed sequence satisfies $(x_1, x_2, \ldots, x_n) \in \mathcal{A}_n$,

and it is *rejected* otherwise. If the alternate hypothesis is rejected the null-hypothesis is maintained. We can then define two kinds of errors:

$$\alpha_n(\mathcal{A}_n) := P_0^{\times n}(\mathcal{A}_n) \tag{3.2}$$
$$\beta_n(\mathcal{A}_n) := 1 - P_1^{\times n}(\mathcal{A}_n) = P_1^{\times n}(\mathcal{A}_n^c) \tag{3.3}$$

The *error of the first kind* or *type-I error*, $\alpha_n(\mathcal{A}_n)$, captures the acceptance of the alternate hypothesis even if the null-hypothesis is true. The *error of the second kind* or *type-2 error*, $\beta_n(\mathcal{A}_n)$, captures the rejection of the alternate hypothesis even though it is true.

Ideally we would like to devise a sequence of tests such that both of these errors are small, and get smaller as $n$ increases. We can compute the optimal average error assuming an uniform (or unbiased) prior on the two distributions.

Can you formulate this problem in the general framework of probability theory as covered in Chapter 0? What is a test in this framework?

For two pmfs $P_0$ and $P_1$ and $n \in \mathbb{N}$, we define the *optimal unbiased average error* with uniform prior as

$$\epsilon^*_{\text{sym},n}(P_0, P_1) := \frac{1}{2} \min_{\mathcal{A}_n \subset \mathcal{X}^n} \left( \alpha_n(\mathcal{A}_n) + \beta_n(\mathcal{A}_n) \right). \tag{3.4}$$

Here uniform prior means that the probability we assign to the two hypotheses prior to observing the random sequence is equal, and thus $\epsilon^*_{\text{sym},n}$ is indeed the probability of making a wrong decision. However, as their names indicates, often these two hypotheses are not treated on the same footing. Indeed, the question can be easily generalised to the case when the prior over the two hypotheses is not uniform. If $p \in (0,1)$ is the probability that $H_0$ is correct, we define

$$\epsilon^*_{p,n}(P_0, P_1) := \min_{\mathcal{A}_n \subset \mathcal{X}^n} \left( p\alpha_n(\mathcal{A}_n) + (1-p)\beta_n(\mathcal{A}_n) \right). \tag{3.5}$$

The probability $p$ is called the prior in this setting. For such cases it is natural to look at a somewhat different and inherently asymmetric formulation of the problem. Namely, we simply require that one of the errors (by convention the error of the first kind) is upper bounded by a constant $\epsilon$ and ask how small we can make the other error.

**Example.** *Assume the alternate hypothesis is that a patient is suffering from COVID-19, and the null-hypothesis is that this is not the case. The error of the first kind is then a false positive and the error of the second kind is a false negative. If we devise a test distinguishing these two hypothesis we are probably more tolerant of false positives than false negatives.*

For two pmfs $P_0$ and $P_1$ and $n \in \mathbb{N}$ and $\epsilon \in (0,1)$, we define

$$\beta^*_n(\epsilon; P_0, P_1) := \min\{\beta_n(\mathcal{A}_n) : \alpha_n(\mathcal{A}_n) \leq \epsilon\}, \tag{3.6}$$

where $\mathcal{A}_n$ runs through all subsets of $\mathcal{X}^n$.

Asymmetric hypothesis testing also allows us to deal with the situation when we do not know the prior probabilities. In that case the sum (or probabilistic mixture) of the two errors does not make sense

and we need to look at the errors independently. We can, however, still ask the question how these two errors trade off against each other. This is done by analysing $\beta_n^*(\epsilon)$, and in particular by looking at its asymptotics for large $n$.

## 3.2 *Total variation distance*

We will need to compare different probabilistic models, which are each defined by a pmf over observations. Hence, we will need to compare distributions to each other and it makes sense to define a metric on the space of pmfs. Different choice are possible here but we will soon see that the total variation distance is especially meaningful in information theory, statistics and cryptography.

The *total variation distance* (tvd) between two pmfs $P$ and $Q$ is

$$\delta_{\text{tvd}}(P,Q) := \frac{1}{2}\sum_{x\in\mathcal{X}} \left|P(x) - Q(x)\right|. \tag{3.7}$$

The relative entropy can also be used to compare pmfs. Why is it not a metric?

The total variation distance vanishes if and only if $P = Q$ and it reaches its maximum 1 when $P$ and $Q$ are orthogonal, that is, when for every $x \in \mathcal{X}$ either $P(x) = 0$ or $Q(x) = 0$. We can alternatively express the tvd using the following variational formulae, which motivate its name.

**Lemma 3.1.** *For any two pmfs $P$ and $Q$, the following relations hold:*

$$\delta_{\text{tvd}}(P,Q) = \max_{\mathcal{A}\subset\mathcal{X}} \left(\sum_{x\in\mathcal{A}} P(x) - Q(x)\right). \tag{3.8}$$

*Proof.* Consider the test

$$\mathcal{A}_* = \{x \in \mathcal{X} : P(x) > Q(x)\} \tag{3.9}$$

that is optimal for the maximisation in (3.8). For this test, we have

$$\max_{\mathcal{A}\subset\mathcal{X}} \left(\sum_{x\in\mathcal{A}} P(x) - Q(x)\right) = \sum_{x\in\mathcal{A}_*} P(x) - Q(x) = \sum_{x\in\mathcal{A}_*} \left|P(x) - Q(x)\right|. \tag{3.10}$$

On the other hand, by normalisation, we also have

$$\sum_{x\in\mathcal{X}} P(x) - Q(x) = 0\,, \tag{3.11}$$

and thus, we have $\sum_{x\in\mathcal{A}_*} P(x) - Q(x) = \sum_{x\in\mathcal{A}_*^c} Q(x) - P(x)$. Hence,

$$\sum_{x\in\mathcal{A}_*} \left|P(x) - Q(x)\right| = \sum_{x\in\mathcal{A}_*^c} \left|P(x) - Q(x)\right|. \tag{3.12}$$

and thus $\sum_{x\in\mathcal{A}_*} \left|P(x) - Q(x)\right| = \frac{1}{2}\sum_{x\in\mathcal{X}} \left|P(x) - Q(x)\right|$. □

The total variation distance satisfies a data-processing inequality.

Verify that the total variation distance is a metric: it is symmetric, it is positive and zero only if the two distributions are equal, and it satisfies the triangle inequality.

**Proposition 3.2** (DPI for tvd). *For any channel $W_{Y|X}$ and any two pmfs $P_X$ and $Q_X$, we have*

$$\delta_{\mathrm{tvd}}(P_X, Q_X) \geq \delta_{\mathrm{tvd}}(P_Y, Q_Y), \tag{3.13}$$

*where the output distribution are given as*

$$P_Y(y) = \sum_{x \in \mathcal{X}} W_{Y|X}(y|x) P_X(x) \text{ and } Q_Y(y) = \sum_{x \in \mathcal{X}} W_{Y|X}(y|x) Q_X(x). \tag{3.14}$$

This can be understood as saying that after we apply a channel $W_{Y|X}$, that is, introduce some noise, the output distributions are generally closer than the input distributions. So in a sense the two distributions have become more difficult to distinguish after applying the channel. We will verify this property in Exercise 3.1.

There are various bounds relating different measures of distinguishability (see also next chapter). Most famous amongst those is probably Pinsker's inequality, which states that (see Exercise 3.4)

$$\delta_{\mathrm{tvd}}(P, Q) \leq \sqrt{\frac{1}{2} D(P\|Q)}\,. \tag{3.15}$$

## 3.3 Optimal hypothesis tests

We will first try to understand the one-shot setting, i.e. we set $n = 1$. Notably, this actually also covers the cases where $n > 1$ in the sense that we can always see the joint distributions $P_0^{\times n}$ and $P_1^{\times n}$ as our two hypotheses. However, in the one-shot setting we do not have any i.i.d. structure to work with, and the latter often allows us to simplify the problem when $n$ is large.

It turns out that the minimal error probability $\epsilon^*_{\mathrm{sym},1}$ is related to the total variation distance, and we will explore this relation first. We can then deduce a much more general property of optimal tests, called the Neyman–Pearson lemma.

### 3.3.1 Hypothesis testing and total variation distance

In the following, we will see that $\epsilon^*_{\mathrm{sym},1}$ can be expressed in terms of the tvd between two pmfs $P_0$ and $P_1$. Recall that the total variational distance is closely related to the 1-norm, i.e., $\delta_{\mathrm{tvd}}(P_0, P_1) = \frac{1}{2}\|P_0 - P_1\|_1$. We can now state the following result for binary hypothesis testing with general priors.

**Proposition 3.3.** *The one-shot optimal average error is given by*

$$\epsilon^*_{\text{sym},1}(P_0, P_1) = \frac{1}{2}\Big(1 - \delta_{\text{tvd}}(P_0, P_1)\Big). \tag{3.16}$$

This gives a clear operational interpretation for the total variation distance, which is a widely used distance measure in statistics. On the one hand, when $P_0 = P_1$ the total variation distance vanishes and the best thing we can do is a random guess. On the other hand, when $P_0$ and $P_1$ are orthogonal, then we can distinguish them perfectly and the error vanishes.

*Proof of Proposition 3.3.* First we observe the following relations:

$$\begin{aligned}
\epsilon^*_{\text{sym},1} &= \frac{1}{2}\min_{\mathcal{A}\subset\mathcal{X}}\big(P_0(\mathcal{A}) + P_1(\mathcal{A}^c)\big) && (3.17)\\
&= \frac{1}{2} - \frac{1}{2}\max_{\mathcal{A}\subset\mathcal{X}}\big(P_0(\mathcal{A}^c) - P_1(\mathcal{A}^c)\big) && (3.18)\\
&= \frac{1}{2} - \frac{1}{2}\delta_{\text{tvd}}(P_0, P_1), && (3.19)
\end{aligned}$$

where we used Proposition 3.1 in the last step. □

**Example.** *Consider a DMS with either $P_1 = (\frac{1}{2}, \frac{1}{2})$ or $P_2 = (\frac{3}{4}, \frac{1}{4})$. For $n = 2$ we use the shorthand notation $00, 01, 10, 11$ to denote the different possible sequences in $\mathcal{X}^2$. The possible threshold tests are:*

| $A$ | $\alpha_2$ | $\beta_2$ | $T$ |
|---|---|---|---|
| $\emptyset$ | $0$ | $1$ | $(-\infty, 2\log\frac{2}{3})$ |
| $\{00\}$ | $\frac{1}{4}$ | $\frac{7}{16}$ | $[2\log\frac{2}{3}, \log\frac{4}{3})$ |
| $\{00, 01, 10\}$ | $\frac{3}{4}$ | $\frac{1}{16}$ | $[\log\frac{4}{3}, 2)$ |
| $\mathcal{X}^2$ | $1$ | $0$ | $[2, +\infty)$ |

*Here $\alpha = \alpha(A)$ and $\beta = \beta(A)$ are two two kinds of errors and we give the range of $T$ which produces this test.*

### 3.3.2 *The Neyman-Pearson lemma*

The test in Eq. (3.9) is of the form

$$\mathcal{A}_* = \left\{x \in \mathcal{X} : \log\frac{P_0(x)}{P_1(x)} \le T\right\}. \tag{3.20}$$

where $T$ is a threshold and $\log\frac{P_0(x)}{P_1(x)}$ is the log-likelihood ratio (LLR) we encountered previously. We can show that optimal tests must always have this form.

**Lemma 3.4** (Neyman-Pearson Lemma)**.** *Let $P_0, P_1$ be two pmfs. For any $T \in \mathbb{R}$, define the region*

$$\mathcal{A}_*(T) = \left\{x : \log\frac{P_0(x)}{P_1(x)} \le T\right\}. \tag{3.21}$$

*Then, for any test $\mathcal{A}$ and any $T \in \mathbb{R}$, the following holds:*

$$\begin{aligned}
\alpha_1(\mathcal{A}) < \alpha_1(\mathcal{A}_*(T)) &\implies \beta_1(\mathcal{A}) > \beta_1(\mathcal{A}_*(T)) && (3.22)\\
\beta_1(\mathcal{A}) < \beta_1(\mathcal{A}_*(T)) &\implies \alpha_1(\mathcal{A}) > \alpha_1(\mathcal{A}_*(T)) && (3.23)
\end{aligned}$$

*Moreover, we can replace the strict inequalities with non-strict inequalities*

*in both implications above. Furthermore, we have*

$$\alpha_1(\mathcal{A}_*) + 2^T \beta_1(\mathcal{A}_*) \leq \alpha_1(\mathcal{A}) + 2^T \beta_1(\mathcal{A}). \tag{3.24}$$

*Proof.* We fix $T$ and write $\mathcal{A}_*(T) = \mathcal{A}_*$. For every $x \in \mathcal{X}$, we have

$$(\mathbf{1}\{x \in \mathcal{A}_*\} - \mathbf{1}\{x \in \mathcal{A}\})\,(P_0(x) - 2^T P_1(x)) \leq 0\,. \tag{3.25}$$

To verify this, note that if $\mathbf{1}\{x \in \mathcal{A}_*\} = 1$ and thus $\mathbf{1}\{x \in \mathcal{A}_*\} - \mathbf{1}\{x \in \mathcal{A}\} \geq 0$ then $P_0(x) - 2^T P_1(x)$ is negative by the definition of $\mathcal{A}_*$, and, vice versa, if $\mathbf{1}\{x \in \mathcal{A}_*\} = 0$ and thus $\mathbf{1}\{x \in \mathcal{A}_*\} - \mathbf{1}\{x \in \mathcal{A}\} \leq 0$ then $P_0(x) - 2^T P_1(x)$ is positive. Now summing over $x \in \mathcal{X}$ we find

$$\alpha_1(\mathcal{A}_*) - \alpha_1(\mathcal{A}) - 2^T\,(1 - \beta_1(\mathcal{A}_*) - (1 - \beta_1(\mathcal{A})) \leq 0\,, \tag{3.26}$$

or, equivalently, $\alpha_1(\mathcal{A}_*) - \alpha_1(\mathcal{A}) \leq 2^T\,(\beta_1(\mathcal{A}) - \beta_1(\mathcal{A}_*))$. Since $2^T > 0$ the implication in Eq. (3.23) follows for both strict and non-strict inequalities follow. The implication in Eq. (3.22) is then the contrapositive of Eq. (3.23).

The last inequality is simply a rewriting of Eq. (3.26). □

## 3.4 *The Chernoff exponent in symmetric hypothesis testing*

When we look at $n$ i.i.d. copies of the sample, distributed according to $P_0^{\times n}$ or $P_1^{\times n}$, respectively, we make the—at first sight surprising—observation that these two distributions get closer and closer to orthogonal as $n \to \infty$ (unless $P_0 = P_1$, of course). Or, in other words, the total variation distance between $P_0^{\times n}$ and $P_1^{\times n}$ converges to 1 as $n \to \infty$. This means that for large $n$ it becomes easier and easier to distinguish the two distributions. We are interested in how fast this convergence is.

### 3.4.1 *The Method of Types*

Consider a sequence $x^n$ where each symbol $x_i$ for $i \in [n]$ is taken from a finite set $\mathcal{X}$. There are obviously $|\mathcal{X}|^n$ different such sequences. But in many circumstances it suffices to classify these sequences simply by how many times each of the elements of $\mathcal{X}$ appears in it. This is called the *type* of the sequence $x^n$ and motivates the "Method of Types"[2], of which we however will only be able to scratch the surface.

We will first introduce the empirical distribution of a sequence $x^n$, which is in one-to-one correspondence with its type.

[2] Imre Csiszár. The Method of Types. *IEEE Transactions on Information Theory*, 44(6):2505–2523, oct 1998. DOI: 10.1109/18.720546

Let $n \in \mathbb{N}$. For any sequence $x^n \in \mathcal{X}^n$, we define a pmf $P_{[x^n]} \in \mathcal{P}(\mathcal{X})$, the *empirical distribution* of $x^n$, for all $x \in \mathcal{X}$ as

$$P_{[x^n]}(x) = \frac{1}{n}\big|\{i \in [n] : x_i = x\}\big|. \tag{3.27}$$

Moreover, we denote by $[x^n]$ the set of all sequences that have the same empirical distribution as $x^n$. The set of all empirical distributions of length $n$ is denoted $\mathcal{T}_n$.

Note that this is a pmf on $\mathcal{X}$ where each symbol has probability proportional to the number of times it appears in $x^n$.

We will see in Exercise 3.3 that the empirical distribution of an i.i.d. random sequence $X^n$ will concentrate around the underlying pmf $P_X$ with which the $X_i$ are chosen.

Here, let us summarise some important properties of types

**Proposition 3.5.** *We have* $|\mathcal{T}_n| \leq (n+1)^{|\mathcal{X}|-1}$. *Moreover,*

$$\frac{1}{|\mathcal{T}_n|}2^{nH(X)} \leq \big|[x^n]\big| \leq 2^{nH(X)}, \tag{3.28}$$

*where $X$ is distributed according to $P_{[x^n]}$.*

*Proof.* The number of different types of sequences in $\mathcal{X}^n$ is simply given by the number of partitions of $n$ into $|\mathcal{X}|$ segments. The cardinality bound on $|\mathcal{T}_n|$ can be shown by counting the number of possible values the probability $P_{[x^n]}(x)$ in Eq. (3.27) can take for each symbol $x \in \mathcal{X}$, and noting that the last probability is determined by the others. This yields $|\mathcal{T}_n| \leq (n+1)^{|\mathcal{X}|-1}$.

To show the upper bound in Eq. (3.28), we note if $x^n$ is i.i.d. following the law $Q$, then

$$\begin{aligned}
\log Q^{\times n}(x^n) &= \sum_{i=1}^{n} \log Q(x_i) && (3.29)\\
&= n \sum_{x\in\mathcal{X}} P_{[x^n]}(x) \log Q(x) && (3.30)\\
&= n\Big( - H(X)_{P_{[x^n]}} - D\big(P_{[x^n]}\big\|Q\big)\Big). && (3.31)
\end{aligned}$$

Hence, the upper bound in Eq. (3.28) follows by taking the special case with $Q = P_{[x^n]}$ in the above, which gives

$$1 \geq \sum_{x^n \in [x^n]} P\big[X^n = x^n\big] = \big|[x^n]\big| \cdot 2^{-nH(X)}. \tag{3.32}$$

The lower bound will be covered in Exercise 3.6. □

One important observation here is that $|\mathcal{T}_n|$ grows only polynomially in $n$, which we will use to our advantage.

### 3.4.2 *Chernoff exponent*

We first want to introduce the Chernoff distance.

The *Chernoff distance* between two pmfs $P_0$ and $P_1$ is defined as

$$C(P_0, P_1) := \max_{\lambda \in [0,1]} -\log \sum_{x \in \mathcal{X}} P_0(x)^\lambda P_1(x)^{1-\lambda} . \tag{3.33}$$

We will need a variational expression for the Chernoff distance, which expresses it in terms of the relative entropy.

**Lemma 3.6.** *The following equation holds:*

$$C(P_0, P_1) = \min_{Q \in \mathcal{P}(\mathcal{X})} \max \left\{ D(Q\|P_0), D(Q\|P_1) \right\} . \tag{3.34}$$

*Proof.* We may use the minimax theorem (cf. Propositon 0.13) to write

$$\begin{aligned} &\min_{Q \in \mathcal{P}(\mathcal{X})} \max \left\{ D(Q\|P_0), D(Q\|P_1) \right\} && (3.35)\\ &= \min_{Q \in \mathcal{P}(\mathcal{X})} \max_{\lambda \in [0,1]} \lambda D(Q\|P_0) + (1-\lambda) D(Q\|P_1) && (3.36)\\ &= \max_{\lambda \in [0,1]} \min_{Q \in \mathcal{P}(\mathcal{X})} \lambda D(Q\|P_0) + (1-\lambda) D(Q\|P_1) . && (3.37) \end{aligned}$$

We then observe that

$$\begin{aligned} \lambda D(Q\|P_0) + (1-\lambda) D(Q\|P_1) &= \sum_{x \in \mathcal{X}} Q(x) \log \frac{Q(x)}{P_0(x)^\lambda P_1(x)^{1-\lambda}} && (3.38)\\ &= D(Q\|P_\lambda) + C(P_0\|P_1), && (3.39) \end{aligned}$$

where we introduced the pmf

$$P_\lambda(y) = \frac{P_0(y)^\lambda P_1(y)^{1-\lambda}}{\sum_{x \in \mathcal{X}} P_0(x)^\lambda P_1(x)^{1-\lambda}} . \tag{3.40}$$

From here it now becomes clear that $Q = P_\lambda$ minimises the expression due to the positive definiteness of the relative entropy, which concludes the proof. □

We are now ready to show that the error exponent for symmetric hypothesis testing is given by the Chernoff distance.

**Theorem 3.7.** *For any two non-orthogonal pmfs $P_0$ and $P_1$, we have*

$$\lim_{n \to \infty} -\frac{1}{n} \log \epsilon^*_{\mathrm{sym},n}(P_0, P_1) = C(P_0, P_1) . \tag{3.41}$$

Note that this implies an asymptotic upper bound on the probability of error, so it says that there exists a sequence of tests for which the error drops as $2^{-nC(P_0,P_1)}$, and that this is optimal.

One may want to verify that $C(P_0, P_1)$ is positive. To do so, first note that it suffices to verify that

$$\sum_x \sqrt{P_0(x) P_1(x)} \leq 1$$

holds. This is however ensured by the Cauchy-Schwarz inequality (Prop. 0.9). Moreover, this ensures that $C(P_0, P_1) \geq 0$ with equality iff $P_0 = P_1$.

*Proof.* We start by noting that

$$2\epsilon^*_{\mathrm{sym},n} = \min_{\mathcal{A}_n \subset \mathcal{X}^n} P_0^{\times n}(\mathcal{A}_n) + P_1^{\times n}(\mathcal{A}_n^c) \tag{3.42}$$

$$= \sum_{x^n \in \mathcal{X}^n} \min\{P_0^{\times n}(x^n), P_1^{\times n}(x^n)\}\,. \tag{3.43}$$

The second equality follows from the fact that the optimal test is a $T = 0$ threshold test, i.e., we include $x^n$ in $\mathcal{A}_n$ iff $P_0^{\times n}(x^n) < P_1^{\times n}(x^n)$.

To show the upper bound on the error probability, we then argue that, for every $\lambda \in [0,1]$:

$$2\epsilon^*_{\mathrm{sym},n} \leq \sum_{x^n \in \mathcal{X}^n} P_0^{\times n}(x^n)^\lambda P_1^{\times n}(x^n)^{1-\lambda} \tag{3.44}$$

$$= \sum_{x_1 \in \mathcal{X}} \dots \sum_{x_n \in \mathcal{X}} P_0(x_1)^\lambda P_1(x_1)^{1-\lambda} \dots P_0(x_n)^\lambda P_1(x_n)^{1-\lambda} \tag{3.45}$$

$$= \left( \sum_{x \in \mathcal{X}} P_0(x)^\lambda P_1(x)^{1-\lambda} \right)^n \tag{3.46}$$

We take the logarithm on both sides and divide through $n$ to get

$$-\frac{1}{n} \log \epsilon^*_{\mathrm{sym},n} \geq -\log \sum_{x \in \mathcal{X}} P_0(x)^\lambda P_1(x)^{1-\lambda} + \frac{1}{n} \tag{3.47}$$

What changes when we do the same analysis for $\epsilon^*_{p,n}$?

The last term vanishes in the limit $n \to \infty$, and thus the bound (3.41) follows by optimising the right-hand side over all $\lambda \in [0,1]$.

To show the lower bound, we realise that the minimum only depends on the type of $x^n$, so we can write

$$2\epsilon^*_{\mathrm{sym},n} = \sum_{[x^n] \in \mathcal{T}_n} \big|[x^n]\big| \min\{P_0^{\times n}(x^n), P_1^{\times n}(x^n)\} \tag{3.48}$$

$$\geq \frac{1}{|\mathcal{T}_n|} \sum_{[x^n] \in \mathcal{T}_n} 2^{nH(X)_{P_{[x^n]}} + \min\{\log P_0^{\times n}(x^n),\, \log P_1^{\times n}(x^n)\}} \tag{3.49}$$

$$= \frac{1}{|\mathcal{T}_n|} \sum_{[x^n] \in \mathcal{T}_n} 2^{-n \cdot \max\{D(P_{[x^n]} \| P_0),\, D(P_{[x^n]} \| P_1)\}} \tag{3.50}$$

$$\geq \frac{1}{|\mathcal{T}_n|} 2^{-n \min_{Q \in \mathcal{T}_n} \cdot \max\{D(Q\|P_0),\, D(Q\|P_1)\}}, \tag{3.51}$$

where the inequality employs the bound on $\big|[x^n]\big|$ from Prop. 3.5 and the penultimate equality is due to Eq. (3.31). Taking the same limit as above, and using the fact that $\frac{1}{n} \log |\mathcal{T}_n| \to 0$ as $n \to \infty$, we find that

$$\lim_{n\to\infty} -\frac{1}{n} \log \epsilon^*_{\mathrm{sym},n} \leq \lim_{n\to\infty} \min_{Q \in \mathcal{T}_n} \cdot \max\{D(Q\|P_0),\ D(Q\|P_1)\}\,. \tag{3.52}$$

Finally, since every element in $\mathcal{P}(\mathcal{X})$ can be approximated arbitrarily well by an element in $\mathcal{T}_n$ for large enough $n$, and the relative entropy is continuous, we can use Lemma 3.6 to conclude the proof.[3] □

[3] This is admittedly a bit rough, but making such continuity statements precise is not worth our time here.

## 3.5 *Stein's lemma in asymmetric hypothesis testing*

For simplicity we assume that $D(P_0\|P_1) < \infty$ in the following, as otherwise by definition of the relative entropy there are some $x \in \mathcal{X}$ with $P_0(x) > 0$ but $P_1(x) = 0$, and, as we will see in the homework, it is possible to come up with tests that have $\beta_n^*(\epsilon) = 0$ for large enough $n$.

Under this assumption, our goal is to show that regardless of the constant upper bound $\epsilon$ on the type-I error, the type-II error behaves as

$$\beta_n^*(\epsilon) \approx 2^{-nD(P_0\|P_1)}, \tag{3.53}$$

where the approximation is up to factors that are sub-exponential in $n$. This means that the optimal exponential rate at which the type-II error approaches zero is determined by the relative entropy (in first order), thus giving the relative entropy $D(P_0\|P_1)$ a clear operational interpretation in statistics. Let us restate this as a theorem:

**Theorem 3.8** (Chernoff-Stein Lemma)**.** *For every $\epsilon \in (0,1)$,*

$$\lim_{n\to\infty} -\frac{1}{n}\log \beta_n^*(\epsilon) = D(P_0\|P_1) \tag{3.54}$$

For the proof we will use the information spectrum method (see[4] for more information on that technique). Consider a random variable $X$ that takes values in $\mathcal{X}$ and two pmfs $P_0, P_1 \in \mathcal{P}(\mathcal{X})$ as above. Recall that the log-likelihood ratio (LLR) for the two pmfs is the random variable

$$Z = \log \frac{P_0(X)}{P_1(X)}, \tag{3.55}$$

where $X$ (and thus $Z$) is distributed according to $P_0$. The log-likelihood ratio is an important random variable in the analysis of many different information processing tasks. We now introduce the following quantity:

[4] T. S. Han. *Information-Spectrum Methods in Information Theory*. Applications of Mathematics. Springer, 2002

Verify that the expectation value of $Z$ under $P_0$ is $D(P_0\|P_1)$.

For two pmfs $P_0, P_1$ and $\epsilon \in (0,1)$ we define the *information-spectrum divergence* as

$$D_s^\epsilon(P_0\|P_1) := \sup\{R \in \mathbb{R} : P_0[Z \le R] \le \epsilon\} \tag{3.56}$$

This quantity looks complicated at first sight, but it simply evaluates exactly where (the value $R$) we need to cut off the pmf for the LLR, $Z$, so that the probability that $Z \le R$ is at most $\epsilon$. One could also simply see it as an inverse of the cumulative distribution function of $Z$. (This result can also be interpreted as a consequence of the Neyman-Pearson lemma, which states that all tests optimising the two types of error are threshold tests for the LLR.)

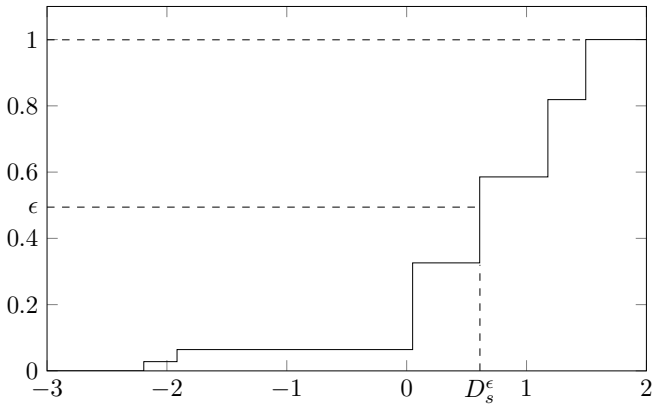



Figure 3.1: Example of the information spectrum. The plot shows the cumulative distribution of the LLR and the value of $D_s^\epsilon(P\|Q)$ for some example distributions.

**Lemma 3.9.** *Let $n \in \mathbb{N}$, $\epsilon \in (0,1)$ and $\delta \in (0, 1-\epsilon)$. The following two inequalities hold:*

$$D_s^\epsilon\left(P_0^{\times n}\middle\|P_1^{\times n}\right) \leq -\log \beta_n^*(\epsilon) \leq D_s^{\epsilon+\delta}\left(P_0^{\times n}\middle\|P_1^{\times n}\right) + \log\frac{1}{\delta}\,. \quad (3.57)$$

For $n = 1$ this gives bounds on asymmetric hypothesis testing for any two distributions $P_0$ and $P_1$, without using the i.i.d. structure. If one plugs in $n$-fold i.i.d. distributions instead this recovers the result for general $n$, and thus it is sufficient to prove it for $n = 1$. This is an example of a *one-shot bound*, a generic bound on an information-theoretic quantity that can then be easily statistically analysed by taking advantage of an i.i.d. or similar structure.

*Proof.* To get the lower bound, we use a threshold test in the sense of Neyman-Pearson (see Lemma 3.4) of the form

$$\mathcal{A}_T := \{x \in \mathcal{X} : P_0(x) \leq 2^T P_1(x)\}\,. \quad (3.58)$$

Let us choose $T = D_s^\epsilon(P_0\|P_1) - \mu$ for some $\mu > 0$ that can be chosen arbitrarily small. The reason we need this small slack $\mu > 0$ is simply that by definition of the supremum in (3.56) this ensures that we have $\alpha_n(\mathcal{A}_T) = P_0(\mathcal{A}_T) \leq \epsilon$ for any $\mu > 0$, while the same might not necessarily be true for $\mu = 0$.[5] Moreover, we have

$$\begin{aligned}
\beta_1(\mathcal{A}_T) &= P_1(\mathcal{A}_T^c) && (3.59)\\
&= \sum_{x\in\mathcal{X}} P_1(x)\,\mathbf{1}\{P_0(x) > 2^T P_1(x)\} && (3.60)\\
&\leq 2^{-T} \sum_{x\in\mathcal{X}} P_0(x)\,\mathbf{1}\{P_0(x) > 2^T P_1(x)\} && (3.61)\\
&\leq 2^{-T}\,. && (3.62)
\end{aligned}$$

[5] Recall the definition of the supremum on sets that are not closed and note that the set of all $R \in \mathbb{R}$ satisfying $P[Z \leq R] \leq \epsilon$ is not closed in general.

This directly implies that $\beta_1^*(\epsilon) \leq 2^{-T}$, or, equivalently,

$$-\log \beta_1^*(\epsilon) \geq D_s^\epsilon(P_0\|P_1) - \mu\,. \tag{3.63}$$

Since this holds for all $\mu > 0$ we get the desired inequality.

To get the upper bound, let $\mathcal{A}_*$ be the optimal test for $\beta_1^*(\epsilon)$, i.e. we have $\alpha_1(\mathcal{A}_*) \leq \epsilon$ and $\beta_1(\mathcal{A}_*) = \beta_1^*(\epsilon)$. With Lemma 3.4, we get

$$\begin{aligned}
\epsilon + 2^R\beta_1^*(\epsilon) &\geq \alpha_1(\mathcal{A}_*) + 2^R\beta_1(\mathcal{A}_*) && (3.64)\\
&\geq \alpha_1(\mathcal{A}_R) + 2^R\beta_1(\mathcal{A}_R) && (3.65)\\
&\geq \alpha_1(\mathcal{A}_R) && (3.66)\\
&= P_0\left[\log\frac{P_0(X)}{P_1(X)} \leq R\right]. && (3.67)
\end{aligned}$$

where we introduced a threshold test $\mathcal{A}_R$ at rate $R$. Choosing $R = \log\delta - \log\beta_n^*(\epsilon)$, the above implies that

$$P_0^{\times n}[Z \leq R] \leq \epsilon + \delta \tag{3.68}$$

and thus we have $D_s^{\epsilon+\delta}(P_0^{\times n}\|P_1^{\times n}) \geq R = -\log\beta_n^*(\epsilon) + \log\delta$, which is the desired upper bound. □

In Exercise 1.6 you have derived the following limit.

**Lemma 3.10.** *Let $P_0, P_1 \in \mathcal{P}(X)$ such that $D(P_0\|P_1) < \infty$ and $\epsilon \in (0,1)$. Then,*

$$\lim_{n\to\infty}\frac{1}{n}D_s^\epsilon(P_0^{\otimes n}\|P_1^{\times n}) = D(P_0\|P_1)\,. \tag{3.69}$$

The limit is essentially a direct consequence of the law of large numbers applied for the random variable $Z = \sum_{i=1}^n \log P_0(X_i) - \log P_1(X_i)$, where $X_i$ are i.i.d. distributed according to the law $P_0$.

As an aside, we can evaluate the quantity on the left, $\frac{1}{n}D_s^\epsilon(P_0^n\|P_1^n)$, even to higher orders in $n$ using the central limit theorem. While we will not need this here, analysing such higher order terms has been a fruitful area of research recently as it allows us to make more precise statements about optimal errors for smaller $n$, and thus for practical settings where we are far from the asymptotic setting of very large $n$.

*Proof of Theorem 3.8.* The proof of the theorem is evident once we combine Lemma 3.9 and Lemma 3.10. Namely, from Lemma 3.9 we get

$$\frac{1}{n}D_s^\epsilon\big(P_0^{\times n}\big\|P_1^{\times n}\big) \leq -\frac{1}{n}\log\beta_n(\epsilon) \leq \frac{1}{n}D_s^{\epsilon+\delta}\big(P_0^{\times n}\big\|P_1^{\times n}\big) + \frac{1}{n}\log\frac{1}{\delta} \tag{3.70}$$

and in the limit $n \to \infty$ both the lower and upper bound converge to the relative entropy by Lemma 3.10. □

## 3.6 Exercises

**Exercise 3.1** (DPI for tvd). *Let $\mathcal{X}$ and $\mathcal{Y}$ be finite sets. Let $P_X, Q_X \in \mathcal{P}(\mathcal{X})$ and let $\{T(y|x)\}_{y\in\mathcal{Y},x\in\mathcal{X}}$ be some conditional pmf. Suppose*

$$P_Y(y) = \sum_{x\in\mathcal{X}} T(y|x)\cdot P_X(x) \quad \textit{and} \quad Q_Y(y) = \sum_{x\in\mathcal{X}} T(y|x)\cdot Q_X(x)$$

*for each $y \in \mathcal{Y}$. Prove that*

$$\delta_{tvd}(P_Y, Q_Y) \leq \delta_{tvd}(P_X, Q_X).$$

**Exercise 3.2.** *Consider the three pmfs $P$ and $Q$, and $S$ given by the probability vectors $p = (\frac{1}{2}, \frac{1}{4}, \frac{1}{4})$, $q = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ and $s = (0, \frac{1}{2}, \frac{1}{2})$, respectively.*

a) *Compute the unbiased (symmetric) error probabilities $\epsilon^*_{\text{sym},1}$ for binary hypothesis testing for all three pairs: $(P, Q)$, $(P, S)$ and $(Q, S)$.*

b) *Consider the problem of ternary hypothesis testing between the three distributions, when all three have equal priors. Find the minimal error probability and an optimal test.*

c) *Compute $D(P\|Q)$ and $D(P\|S)$.*

d) *Consider asymmetric hypothesis testing for $P$ and $S$. For each $\epsilon > 0$, find $N_0(\epsilon) \in \mathbb{N}$ such that $\beta_n(\epsilon) = 0$ for all $n \geq N_0(\epsilon)$. What does that say about the limit $\lim_{n\to\infty} -\frac{1}{n} \log \beta_n(\epsilon)$? Interpret your result in item c) in this light.*

**Hint:** Note that there are now more than two types of errors and there is no closed formula for the minimal error probability. You have to construct the optimal test manually, for example by trial and error.

**Exercise 3.3** (Empirical typical set)**.** *Let $P \in \mathcal{P}(\mathcal{X})$ be a pmf over some finite set $\mathcal{X}$. The empirical typical set (of length $n$ and tolerance $\epsilon$) w.r.t. $P$ is defined as*

$$\mathcal{A}^{(n)}_{\text{emp},\epsilon}(P) := \left\{ x^n \in \mathcal{X}^n : \delta_{tvd}(P_{[x^n]}, P) \leq \epsilon \right\},$$

*where $P_{[x^n]}$ is the empirical distribution induced by $x^n$.*

*Prove that, for any $\epsilon \in (0, 1]$,*

$$\lim_{n\to\infty} P\left[ X^n \in \mathcal{A}^{(n)}_{\text{emp},\epsilon}(P_X) \right] = 1.$$

**Exercise 3.4** (Pinsker's inequality)**.** *For two pmfs $P$ and $Q$, we want to show that*

$$\delta_{\text{tvd}}(P, Q) \leq \sqrt{\frac{1}{2} D(P\|Q)}\,.$$

a) *Let $f(x) = p \log x + (1 - p) \log(1 - x)$. Show that for $x \in [q, p]$,*

$$f'(x) \geq 4(p - x),$$

*and, thus, $f(p) - f(q) \geq 2(p - q)^2$.*

b) *For two Bernoulli pmfs $\tilde{P}$ and $\tilde{Q}$ defined by $p$ and $q$ respectively, prove Pinsker's inequality*

$$\delta_{\text{tvd}}(\tilde{P}, \tilde{Q}) \leq \sqrt{\frac{1}{2} D(\tilde{P}\|\tilde{Q})}.$$

c) *Let $P$ and $Q$ be two general pmfs. Define $\mathcal{S} = \{x \in \mathcal{X} : P(x) \geq Q(x)\}$. Let $\tilde{P}$ and $\tilde{Q}$ be two Bernoulli pmfs defined by $p = \sum_{x\in\mathcal{S}} P(x)$ and $q = \sum_{x\in\mathcal{S}} Q(x)$ respectively. Show that*

$$\delta_{\text{tvd}}(P, Q) = \delta_{\text{tvd}}(\tilde{P}, \tilde{Q}) \qquad \textit{and} \qquad D(\tilde{P}\|\tilde{Q}) \leq D(P\|Q)\,.$$

**Exercise 3.5** (Continuity of entropy). *For $\epsilon \in [0, \frac{1}{2}]$ and two distributions $P, Q \in \mathcal{P}(\mathcal{X})$ on a finite set $\mathcal{X}$ such that $\delta_{tvd}(P, Q) \leq \epsilon$, show that*

$$\left|H(X)_P - H(X)_Q\right| \leq h(\epsilon) + \epsilon \log |\mathcal{X}|.$$

*where $h : t \mapsto -t \log t - (1-t) \log (1-t)$ is the binary entropy.*

**Hint:** Construct a pair of joint random variables $(X_1, X_2)$ with joint distribution $R_{X_1X_2}$ and marginals $R_{X_1} = P_X$, $R_{X_2} = Q_X$ such that $P[X_1 = X_2]$ is large. Then use Fano's inequality.

**Exercise 3.6.** *Here we show that the set of sequences of type $[x^n]$ satisfies*

$$\left|[x^n]\right| \geq \frac{1}{|\mathcal{T}_n|} 2^{nH(X)},$$

*where $X$ is distributed according to $P_{[x^n]}$.*

a) *Let $X^n$ be $n$ i.i.d. random variables sampled according to $Q(x)$. Let $P_{[x^n]}$ be the empirical distribution of $x^n$. Show that*

$$P[X^n = x^n] = 2^{-n(H(X)_{P_{[x^n]}} + D(P_{[x^n]}||Q))},$$

*where $H(X)_{P_{[x^n]}}$ is evaluated for the pmf $P_{[x^n]}$.*

b) *Let $Q(x) = P_{[x^n]}(x)$ and $X^n$ as above. For an arbitrary $[\hat{x}^n]$, we define*

$$x_{\max} = \operatorname*{argmax}_x P_{[\hat{x}^n]}(x) - P_{[x^n]}(x), \; x_{\min} = \operatorname*{argmin}_x P_{[\hat{x}^n]}(x) - P_{[x^n]}(x),$$

*and construct a new type class $[\tilde{x}^n]$ such that*

$$P_{[\tilde{x}^n]}(x) = \begin{cases} P_{[\hat{x}^n]}(x_{\max}) - \frac{1}{n}, & x = x_{\max} \\ P_{[\hat{x}^n]}(x_{\min}) + \frac{1}{n}, & x = x_{\min} \\ P_{[\hat{x}^n]}(x), & \text{otherwise} \end{cases}$$

*Show that*

$$\frac{P[X^n \in [\tilde{x}^n]]}{P[X^n \in [\hat{x}^n]]} = \frac{P_{[x^n]}(x_{\min}) P_{[\hat{x}^n]}(x_{\max})}{P_{[x^n]}(x_{\max})(P_{[\hat{x}^n]}(x_{\min}) + \frac{1}{n})}.$$

c) *Show that $P[X^n \in [x^n]] = \max_{[\hat{x}^n]} P[X^n \in [\hat{x}^n]]$ and use this to derive the desired inequality.*

# 4
# *Cryptography: Randomness extraction*

**Intended learning outcomes:**

- You can check how close a distribution is to a uniformly random and independent distribution.
- You can compute guessing probability and conditional min-entropy.
- You know how to construct a randomness extractor using a family of hash functions.
- You understand that deterministic functions cannot increase entropy.

## *4.1 Problem setup and definitions*

One of the most prominent concepts in cryptography is randomness, and it lies at the core of information-theoretic security. To understand, for example, whether a given bit string is *random*, we do not want to look at a particular instance of the string (although that is interesting as well and leads ultimately to the notion of algorithmic randomness) but instead want to check that the process that created the bit string selected it at random. Similarly and maybe even more evidently, the concept of a *secret* bit string cannot be defined unless we look at the process by which the bit string is produced. If the random process is such that the bit string is independent of any side information held by an eavesdropper, then secrecy (relative to that eavesdropper) can be claimed.

### *4.1.1 Randomness*

Before we can discuss randomness and privacy we need to fix a way to measure the distance between two distributions — the one that we actually achieve and a truly random one. It turns out that due to its statistical interpretation (see previous chapter), the total variation distance is most suitable for this. This is because if the pmfs are close in tvd, this means that they cannot be distinguished easily by any statistical test. So, if a distribution is very close in tvd to a uniform distribution, it cannot easily be distinguished from such a

truly random source.

In the following we thus say that a random variable $Z$ on an alphabet $\mathcal{Z}$ is close to uniformly random if it pmf is close to a uniform pmf in tvd, i.e., if

$$\delta_{\text{tvd}}(P_Z, U_Z) = \frac{1}{2} \sum_{z \in \mathcal{Z}} |P_Z(z) - U_Z(z)| \tag{4.1}$$

is small, where $U_Z$ denotes the uniform distribution on $\mathcal{Z}$.

We can extend the definition of uniformity to the case where some side information $Y$ on $Z$ is available, and we want to make sure that the randomness is in fact not only uniform but also independent of the side information. This leads us to the following more general definition.

Let $P_{ZY}$ be a joint pmf of two random variables $Z$ on $\mathcal{Z}$ and $Y$ on $\mathcal{Y}$. For any $\epsilon \in (0,1)$, we say that $Z$ is *$\epsilon$-close to uniformly random and independent of $Y$* if

$$\delta_{\text{tvd}}(P_{ZY}, U_Z \times P_Y) \leq \epsilon \,. \tag{4.2}$$

Here, it is worth noting that the tvd can be simplified to

$$\delta_{\text{tvd}}(P_{ZY}, U_Z \times P_Y) = \frac{1}{2} \sum_y P_Y(y) \sum_z |P(z|y) - U(z)| \,, \tag{4.3}$$

and thus what we really require is that $P(\cdot|y)$ is close to uniform in expectation over $y$.

### 4.1.2 *Randomness extractors*

We will now consider the task of *randomness extraction*, namely the task of creating approximately uniform and independent random variables from a random source $X$ that is generally neither uniform nor independent of $Y$. In cryptography the i.i.d. assumption (as it appears, for example, in memoryless sources) is often not very natural since we often cannot guarantee that a random source is exactly memoryless. And as cryptographers we tend to always assume the worst, so assumptions might very well be violated. Hence, we want to ensure that our randomness extraction scheme works even if we do not make any assumptions on the structure of the source. See Figure 4.1 for a schematic.

This is generally difficult: one thing we can immediately notice is that if one output of the source is very likely, for example if it appears exactly with probability 0.5, then we can produce exactly one bit of perfect randomness from this source (the new uniform random variable would be the indicator function for the event that

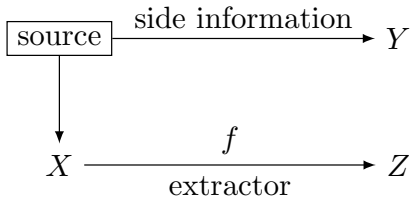



Figure 4.1: The setup of randomness extraction. A source produces random variables $X$ and $Y$, where the latter variable $Y$ is considered as side information on $X$. An extractor $f$ is used to create a new random variable $Z$ that is (close to) uniform and independent of $Y$. An important special case occurs when $Y$ is trivial and we do not have any side information.

this output appears, which takes the value 0 and 1 with probability 0.5 each.), and this is in fact the best we can hope for. The maximal probability over any output of the source thus appears prominently in the analysis of randomness extraction, even in the approximate case, and we will introduce it formally in the next section in terms of guessing probability and min-entropy.

Let us now formally define a randomness extractor for a fixed source, which takes $X$ and produces a bit string $Z$ that is uniformly random and independent of $Y$.

An $(\epsilon, 2^L)$-extractor for a source $X$ with side information $Y$ governed by a pmf $P_{XY}$ is a function $f : \mathcal{X} \to \{0,1\}^L$ such that

$$\delta_{\text{tvd}}(P_{ZY}, U_Z \times P_Y) \leq \epsilon \qquad \text{where} \qquad Z = f(X) \tag{4.4}$$

and thus $P_{ZY}$ is the distribution induced by $f$, i.e.

$$P_{ZY}(z,y) = \sum_{x : f(x)=z} P_{XY}(x,y)\,. \tag{4.5}$$

We may now ask for the maximum length $L$ of such an approximately uniform and independent string of bits. For this purpose we define

$$L^*_\epsilon(X|Y)_P := \max\left\{L \in \mathbb{N} : \exists \text{ an } (\epsilon, 2^L)\text{-extractor for } P_{XY}\right\}. \tag{4.6}$$

We will now find bounds on this quantity from above and below that hold for arbitrary distributions $P_{XY}$. These bounds will be in terms of the smooth min-entropy of the source, which we will introduce in the next section. In the homework we will also consider the special case where these sources are memoryless.

**Example.** *Consider a joint distribution $P_{XY}$ given by the following table:*

| $P_{XY}$ | $x=1$ | $x=2$ | $x=3$ |
|---|---|---|---|
| $y=1$ | $\frac{1}{12}$ | $\frac{1}{4}$ | $\frac{1}{6}$ |
| $y=2$ | $\frac{1}{6}$ | $\frac{1}{4}$ | $\frac{1}{12}$ |

*Clearly $Y$ contains information about $X$; we can in fact compute $H(X) = \frac{3}{2}$ and*

$$I(X:Y) = \frac{3}{2} - \frac{2}{3} - \frac{1}{2}\log 3 > 0\,.$$

*Nonetheless, there is a strategy $f$ to extract one perfect secret bit from $X$ by mapping $\{1,3\} \mapsto 0$ and $2 \mapsto 1$. Then, for $Z = f(X)$ we find*

$$P_{Z|Y}(\cdot|1) = P_{Z|Y}(\cdot|2) = \left(\frac{1}{2}, \frac{1}{2}\right).$$

*Hence, $Z$ is not correlated to $Y$.*

Verify that the above extractor works by checking the tvd condition.

Why should we not allow random functions/channels as extractors here?

## 4.2 *Guessing probability and min-entropy*

We again consider a joint pmf $P_{XY}$ on two random variables $X$ on $\mathcal{X}$ and $Y$ on $\mathcal{Y}$. We characterise our source using the concepts of guessing probability and min-entropy. The guessing probability of $X$ given $Y$ is the probability that an observer with access to $Y$ can correctly guess the value of $X$. It is not difficult to find the optimal

strategy for this task: given a sample $y \in \mathcal{Y}$, the observer will simply choose its guess as

$$\hat{x} = \operatorname*{argmax}_{x \in \mathcal{X}} P_{X|Y}(x|y)\,. \tag{4.7}$$

The average probability of guessing the correct value of $X$ is thus given by the guessing probability as defined in the following.

Let $P_{XY}$ be a joint pmf as above. The *guessing probability* of $X$ conditioned on $Y$ is defined as

$$p_{\text{guess}}(X|Y)_P := \sum_{y \in \mathcal{Y}} P_Y(y) \max_{x \in \mathcal{X}} P_{X|Y}(x|y)\,. \tag{4.8}$$

Moreover, the *conditional min-entropy* of $X$ conditioned on $Y$ is

$$H_{\min}(X|Y)_P := -\log p_{\text{guess}}(X|Y)_P\,. \tag{4.9}$$

The min-entropy belongs to a class of Rényi entropies[1] that have found widespread use in information theory, and we will explore that connection Exercise 4.2. For now let us just point out that it is always smaller than the Shannon entropy.

**Lemma 4.1.** *For any joint pmf $P_{XY}$, we have $H_{\min}(X|Y) \leq H(X|Y)$.*

*Proof.* To see this, we use Jensen's inequality on the convex function $t \mapsto -\log t$ to find

$$\begin{aligned} H_{\min}(X|Y) &= -\log\left(\sum_{y \in \mathcal{Y}} P_Y(y) \max_{x \in \mathcal{X}} P_{X|Y}(x|y)\right) && (4.10)\\ &\leq \sum_{y \in \mathcal{Y}} P_Y(y) \min_{x \in \mathcal{X}} \Big(-\log P_{X|Y}(x|y)\Big) && (4.11)\\ &\leq \sum_{y \in \mathcal{Y}} P_Y(y) \sum_{x \in \mathcal{X}} P_{X|Y}(x|y)\Big(-\log P_{X|Y}(x|y)\Big) = H(X|Y)\,, && (4.12) \end{aligned}$$

where for the second inequality we used the fact that the minimum over $x$ is smaller than the expectation over $x$ under $P_{X|Y}$. □

We will state our results using a variation of the min-entropy, the *smooth min-entropy*, which is the maximum of the min-entropy over a set of distributions that are close to $P_{XY}$ in total variation distance.

Let $P_{XY}$ a joint pmf and $\epsilon \in [0,1)$. We define the $\epsilon$-smooth min-entropy of $X$ conditioned on $Y$ as

$$H^{\epsilon}_{\min}(X|Y)_P := \max_{\widetilde{P}_{X|Y}:\ \delta_{\text{tvd}}(\widetilde{P}_{XY}, P_{XY}) \leq \epsilon} H_{\min}(X|Y)_{\widetilde{P}}\,. \tag{4.13}$$

**Example.** *Consider a source with joint probability distribution as in the previous example. We already exhibited a strategy that can extract a single secret bit. This is in fact optimal (as we will see) since for this distribution it is easy to compute that $H_{\min}(X|Y) = 1$. It is also worth noting that $H(X|Y) > 1$ in this case, but Shannon entropy is not the correct measure to decide how many secret bits we can extract.*

[1] A. Rényi. On Measures of Information and Entropy. In *Proc. 4th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 547–561, Berkeley, California, USA, 1961. University of California Press

where $\widetilde{P}_{XY}(x,y) = P_Y(y)\widetilde{P}_{X|Y}(x|y)$.

We can relate the smooth entropy to the Shannon entropy again if we consider a memoryless source $(\boldsymbol{X}, \boldsymbol{Y})$ producing sequences $X^n$ and $Y^n$. In that case we have the following relation, of which you will prove a special case in Exercise 4.5:

$$\forall \epsilon \in (0,1): \quad \lim_{n\to\infty} \frac{1}{n} H_{\min}^{\epsilon}(X^n|Y^n) = H(X|Y)\,. \tag{4.14}$$

## 4.3 *Achievability via two-universal hash functions*

There are several ways to construct extractors, including using the property of typical sets that all its elements are approximately equally likely. Here we follow a different approach (which is quite standard in cryptography) and use a random construction based on hash functions.

### 4.3.1 *Two-universal hash functions*

In particular, we consider a family of two-universal hash functions $\{f_s\}_{s\in\mathcal{S}}$ where $f_s : \mathcal{X} \to \{0,1\}^L$. They are parametrised by a seed $s$ and have the property that

$$\sum_{s\in\mathcal{S}} \frac{1}{|\mathcal{S}|} \mathbf{1}\{f_s(x) = f_s(x')\} \leq \frac{1}{2^L} \qquad \forall\, x \neq x'\,. \tag{4.15}$$

This condition can equivalently be expressed as

$$P\big[f_S(x) = f_S(x')\big] \leq \frac{1}{2^L} \qquad \forall\, x \neq x'\,. \tag{4.16}$$

where $S$ is distributed uniformly over $\mathcal{S}$. This is the behaviour we expect from a function that produces completely random output when we take a uniformly random seed $s$. Such families of hash functions exist if we choose $\mathcal{S}$ large enough. An example of such a construction is given in Exercise 5.5 using finite-field arithmetic.

Can you nonetheless come up with such a family for the case where $X, Z \in \{0,1\}^L$ as well? The finite fields introduced later in these notes might be helpful!

### 4.3.2 *The $\chi^2$ divergence*

The total variational distance is closely related to the 1-norm distance, which is defined for any vectors $v$ and $w$ that do not necessary need to be normalised. Recall its definition from Section 0.5:

$$\|v - w\|_1 = \sum_{x\in\mathcal{X}} \big|v_x - w_x\big|\,. \tag{4.17}$$

Hence, in particular, $\delta_{\text{tvd}}(P,Q) = \frac{1}{2}\|P - Q\|_1$. We will need the following relation, which is a consequence of the Cauchy-Schwarz inequality from Section 0.5.

**Lemma 4.2.** *Let $P$ and $Q$ be two pmfs such that $Q$ has full support. We have*

$$\delta_{\text{tvd}}(P,Q) \leq \frac{1}{2}\sqrt{\chi^2(P,Q)} \tag{4.18}$$

*where*

$$\chi^2(P,Q) := \sum_x \frac{(P(x)-Q(x))^2}{Q(x)} = \sum_x \frac{P(x)^2}{Q(x)} - 1 \tag{4.19}$$

*is the $\chi^2$-divergence of $P$ with regards to $Q$.*

*Proof.* We use the Cauchy-Schwarz inequality in Lemma 0.10 to bound

$$\begin{aligned} 2\delta_{\text{tvd}}(P,Q) &= \|P-Q\|_1 && (4.20)\\ &= \left\|(P-Q)Q^{-\frac{1}{2}}Q^{\frac{1}{2}}\right\|_1 && (4.21)\\ &\leq \left\|(P-Q)Q^{-\frac{1}{2}}\right\|_2 \left\|Q^{\frac{1}{2}}\right\|_2 = \sqrt{\chi^2(P,Q)}, && (4.22) \end{aligned}$$

where, in the last step, we use the fact that $\sum_x Q(x) = 1$ to verify that the second norm equals 1. Note that in this proof all powers and multiplications of vectors are element-wise. □

### 4.3.3 *The Leftover Hash Lemma*

Let us know apply a function $f_S$ from a two-universal family of hash functions for $S \in \mathcal{S}$ chosen uniformly at random to get an output $Z = f_S(X)$. The following main technical result is also known as the Leftover Hash Lemma[2].

[2] R. Impagliazzo, L. A. Levin, and M. Luby. Pseudo-random generation from one-way functions. In *Proc. ACM STOC 1989*, pages 12–24. ACM Press, 1989. ISBN 0897913078. DOI: 10.1145/73007.73009

**Theorem 4.3.** *Let $\{f_s\}_s$ be a two-universal family of hash functions. Using the notation introduced above, we have*

$$\mathbb{E}\left[\delta_{\text{tvd}}(P_{ZY}^S, U_Z \times P_Y)\right] \leq \frac{1}{2}\sqrt{2^{L-H_{\min}(X|Y)}}, \tag{4.23}$$

*where $P_{ZY}^s(z,y) = \sum_{x\in\mathcal{X}:f_s(x)=z} P_{XY}(x,y)$ and the expectation is taken over a uniformly distributed seed $S$ in $\mathcal{S}$.*

*Proof.* Without loss of generality we can assume that the marginal $P_Y$ has full support as otherwise we can just remove unused symbols.

Let us now first rewrite the left-hand side as

$$\mathbb{E}\left[\delta_{\text{tvd}}(P_{ZY}^S, U_Z \times P_Y)\right] = \mathbb{E}\left[\delta_{\text{tvd}}(P_{Z|Y}^S(\cdot|Y), U_Z)\right] \tag{4.24}$$

where the expectation on the right-hand side is over both $Y$ and $S$.

We then use Lemma 4.2 to further bound

$$\mathbb{E}\left[\delta_{\text{tvd}}(P^S_{ZY}, U_Z \times P_Y)\right] \leq \mathbb{E}\left[\frac{1}{2}\sqrt{\chi^2(P^s_{Z|Y}(\cdot|Y), U_Z)}\right] \tag{4.25}$$

$$= \frac{1}{2}\mathbb{E}\left[\sqrt{2^L g(Y,S) - 1}\right] \tag{4.26}$$

$$\leq \frac{1}{2}\sqrt{2^L \mathbb{E}\left[g(Y,S)\right] - 1}\,, \tag{4.27}$$

where we used Jensen's inequality in the last step to take the expectation under the square root and introduced the function $g(y,s) := \sum_z P^s_{Z|Y}(z|y)^2$. It can be rewritten as

$$g(y,s) = \sum_z \sum_{x,x'} \mathbf{1}\{f_s(x) = z\}\mathbf{1}\{f_s(x') = z\} P_{X|Y}(x|y) P_{X|Y}(x'|y) \tag{4.28}$$

$$= \sum_{x,x'} \mathbf{1}\{f_s(x) = f_s(x')\} P_{X|Y}(x|y) P_{X|Y}(x'|y) \tag{4.29}$$

It remains to analyse the expectation of $g$. We treat the cases where $x \neq x'$ and where $x = x'$ distinctly. In the first case we can apply the property of two-universal hash functions in (4.15). This yields the following bound:

$$\mathbb{E}[g(Y,S)] = \sum_{x,x'} \mathbb{E}[P_{X|Y}(x|Y) P_{X|Y}(x'|Y)] \mathbb{E}[\mathbf{1}\{f_S(x) = f_S(x')\}] \tag{4.30}$$

$$\leq 2^{-L} \sum_{x \neq x'} \mathbb{E}[P_{X|Y}(x|Y) P_{X|Y}(x'|Y)] + \sum_x \mathbb{E}[P_{X|Y}(x|Y) P_{X|Y}(x|Y)] \tag{4.31}$$

$$\leq 2^{-L} + \sum_y P_Y(y) \max_x P_{X|Y}(x|y) \tag{4.32}$$

$$= 2^{-L} + p_{\text{guess}}(X|Y)\,. \tag{4.33}$$

Here, to get the last inequality we simply completed the sum to all $x, x'$ in the first term and then used that $\sum_x P_{X|Y}(x|y) = 1$. Similarly, for the second term, we bounded one of the $P_{X|Y}(x|Y)$ with $\max_x P_{X|Y}(x|Y)$ so that the sum can be computed. Finally, plugging this into Eq. (4.27), we arrive at the desired bound. □

We can leverage this to arrive at the following result.

**Theorem 4.4.** *Consider a source with pmf $P_{XY}$ and let $\epsilon \in (0,1)$. If*

$$L \leq H^{\epsilon-\delta}_{\min}(X|Y) - 2\log\frac{1}{2\delta} \tag{4.34}$$

*for any $\delta \in (0, \epsilon)$, then there exists an $(\epsilon, 2^L)$-extractor for $P_{XY}$. This*

*implies, in particular, that*

$$L^*_\epsilon(X|Y)_P \geq \sup_{\delta\in(0,\epsilon)} H^{\epsilon-\delta}_{\min}(X|Y) - 2\log\frac{1}{\delta} + 1\,. \tag{4.35}$$

*Proof.* Let $\widetilde{P}_{X|Y}$ denote the distribution that achieves the maximum for the smooth min-entropy, i.e. $H^{\epsilon-\delta}_{\min}(X|Y)_P = H_{\min}(X|Y)_{\widetilde{P}}$. Theorem 4.3 applied for the source $\widetilde{P}_{XY}$ with the above choice of $L$ yields

$$\mathbb{E}\left[\delta_{\text{tvd}}(\widetilde{P}^s_{ZY}, U_Z \times P_Y)\right] \leq \delta\,. \tag{4.36}$$

Hence, there is at least one seed value $s$ for which this bound holds, and it remains to show that $f_s$ constitutes an $(\epsilon, 2^L)$-extractor. However, $\delta_{\text{tvd}}(\tilde{P}_{XY}, P_{XY}) \leq \epsilon - \delta$ implies $\delta_{\text{tvd}}(\tilde{P}^s_{ZY}, P^s_{ZY}) \leq \epsilon - \delta$ by the DPI. And hence, using the triangle inequality we have

$$\delta_{\text{tvd}}(P^s_{ZY}, U_Z \times P_Y) \leq \delta_{\text{tvd}}(\widetilde{P}^s_{ZY}, P^s_{ZY}) + \delta_{\text{tvd}}(\widetilde{P}^s_{ZY}, U_Z \times P_Y) \leq \epsilon\,. \tag{4.37}$$

□

## 4.4 Converse via an entropy inequality

The converse relies on the fact that applying a function to a random variable cannot increase the uncertainty about it. We will make this statement formal first and then proof an upper bound on the extractable randomness using it.

### 4.4.1 Functions cannot increase entropy

**Lemma 4.5.** *Let $f : \mathcal{X} \to \mathcal{Z}$ be a function and $P_{XY}$ a pmf. Then*

$$H(X) \geq H(f(X)) \text{ and } H_{\min}(X) \geq H_{\min}(f(X))\,. \tag{4.38}$$

It is really important that in the statement we only allow for deterministic functions, as otherwise the statement does not hold because we can arbitrarily smuggle in entropy through use of a random function.

Give an example where the inequality is violated by a probabilistic function.

*Proof.* Let $Z = f(X)$. We will present the proof for the unconditional case first. The joint distribution $P_{XZ}(x,z) = P_X(x)\,1\{f(x) = z\}$ has the property that $Z$ is deterministic as a function of $X$, and

$$H(XZ) = H(X) + H(Z|X) = H(X)\,. \tag{4.39}$$

Hence, we conclude that $H(X) = H(XZ) = H(Z) + H(X|Z) \geq H(Z)$.

The proof for the min-entropy cannot rely on the chain rule but by inspecting the definition of the respective guessing probabilities,

$$p_{\text{guess}}(X) = \max_x P_X(x) \qquad \text{and} \tag{4.40}$$

$$p_{\text{guess}}(Z) = \max_z P_Z(z) = \max_z \sum_{x:f(x)=z} P_X(x), \tag{4.41}$$

we see that the second term is always at least as large as the first one, i.e., $p_{\text{guess}}(Z) \geq p_{\text{guess}}(X)$. This coincides with our intuition that the input of a function is at least as hard to guess as its output, since once you guessed the input you can get the output by just applying the function. The relation for the min-entropy then follows. □

For our argument we need something similar to the above lemma, but for smooth min-entropy and with side information. Here again we can intuitively argue that it is at least as difficult to guess the input of a function as it is to guess the output (with equality if the function is bijective). Formally, we can show the following:

**Lemma 4.6.** *Let $\epsilon \in [0,1)$ and $f : \mathcal{X} \to \mathcal{Z}$ a surjective function. Then,*

$$H_{\min}^{\epsilon}(X|Y) \geq H_{\min}^{\epsilon}(f(X)|Y)\,. \tag{4.42}$$

*Proof.* The function $f$ can be interpreted as a channel, $W_{Z|X}(z|x) = \delta_{z,f(x)}$ and clearly $Z - X - Y$ form a Markov chain.

We can define an inverse channel, $\widetilde{W}_{X|ZY}$, that recovers the distribution $P_{X|Y}$ by Bayes' rule:

$$\widetilde{W}_{X|ZY}(x|z,y) = \frac{P_{XZ|Y}(x,z|y)}{P_{Z|Y}(z|y)} = \frac{\delta_{z,f(x)} P_{X|Y}(x|y)}{\sum_{x':f(x')=z} P_{X|Y}(x'|y)} \tag{4.43}$$

Since this channel only maps $z$ to values of $x$ with $f(x) = z$ it is in fact a proper right-inverse of $W_{Z|X}$ in the following sense. Let us assume that the distribution $Q_{Z|Y}$ is optimal for the smooth min-entropy $H_{\min}^{\epsilon}(Z|Y)_P$, i.e., $H_{\min}^{\epsilon}(Z|Y)_P = H_{\min}(Z|Y)_Q$. We can then construct

Can you see what goes wrong here if the function is not surjective?

$$Q_{X|Y}(x|y) = \sum_{z'} \widetilde{W}_{X|YZ}(x|z',y) Q_{Z|Y}(z'|y)\,. \tag{4.44}$$

Note now that the pdf $Q_{Z|Y}$ is recovered by applying the function $f$ on the register $X$, i.e., for all $z, y$, it holds that

$$\sum_{x'} W_{Z|X}(z|x') Q_{X|Y}(x'|y) \tag{4.45}$$

$$= \frac{\sum_{x',z'} \delta_{z,f(x')} \delta_{z',f(x')} P_{X|Y}(x'|y) Q_{Z|Y}(z',y)}{\sum_{x':f(x')=z} P_{X|Y}(x'|y)} = Q_{Z|Y}(z|y)\,. \tag{4.46}$$

By the DPI for the TVD we have $\delta_{\text{tvd}}(Q_{XY}, P_{XY}) \leq \delta_{\text{tvd}}(Q_{ZY}, P_{ZY}) \leq \epsilon$. Hence,

$$H^{\epsilon}_{\min}(X|Y)_P \geq H_{\min}(X|Y)_Q = -\log p_{\text{guess}}(X|Y)_Q \,. \tag{4.47}$$

Now we simply use Lemma 4.5 to show that

$$p_{\text{guess}}(X|Y)_Q = \sum_y P_Y(y)\, p_{\text{guess}}(X)_{Q^y} \tag{4.48}$$

$$\leq \sum_y P_Y(y)\, p_{\text{guess}}(Z)_{Q^y} = p_{\text{guess}}(Z|Y)_Q \,, \tag{4.49}$$

where $Q^y_X(x) = Q_{X|Y}(x|y)$ and $Q^y_Z(z) = Q_{Z|Y}(z|y)$, respectively. Combining this with Eq. (4.47) yields the desired bound:

$$H^{\epsilon}_{\min}(X|Y)_P \geq H_{\min}(Z|Y)_Q = H^{\epsilon}_{\min}(Z|Y)_P \,. \tag{4.50}$$

□

### 4.4.2 *Fundamental limit on extractable randomness*

Now we are ready to provide an upper bound on the amount of randomness that can be extracted from a source. It matches the lower bound that we derived using two-universal hash functions, and thus we know that this construction was essentially optimal.[3]

**Theorem 4.7.** *Consider $\epsilon \in (0,1)$ and a source with pmf $P_{XY}$. Then, any $(\epsilon, 2^L)$-extractor for $P_{XY}$ must satisfy*

$$L \leq H^{\epsilon}_{\min}(X|Y)_P \tag{4.51}$$

*Or, in other words, we have $L^*_{\epsilon}(X|Y)_P \leq H^{\epsilon}_{\min}(X|Y)_P$.*

*Proof.* Let us assume there exists a function $f$ that constitutes an $(\epsilon, 2^L)$-extractor. We then necessarily have

$$\delta_{\text{tvd}}(P_{ZY}, U_Z \times P_Y) \leq \epsilon \tag{4.52}$$

for $Z = f(X)$ and $P_Y$ naturally unchanged by the extractor. Hence,

$$H^{\epsilon}_{\min}(Z|Y)_P \geq H_{\min}(Z|Y)_{U\times P} = H_{\min}(Z)_U = L \,, \tag{4.53}$$

where we simply evaluated the min-entropy for the distribution $U_Z \times P_Y$, which is $\epsilon$-close to the distribution $P_{ZY}$. Combining this with Lemma 4.6 yields the bound $H^{\epsilon}_{\min}(X|Y)_P \geq L$, and since this holds for any $(\epsilon, 2^L)$-extractor we have shown the desired statement. □

[3] To be more precise, by essentially optimal we meant that if we apply both the achievability and converse bounds to a DMS which produces independent samples from the distribution $P_{XY}$, then our two bounds from Theorems 4.4 and 4.7 asymptotically coincide, i.e. we can use (4.14) to establish that

$$\lim_{n\to\infty} \frac{1}{n} L^*_{\epsilon}(X^n|Y^n)_P = H(X|Y) \,.$$

## 4.5 Exercises

**Exercise 4.1.** *It is possible to construct distributions that have a large gap between min-entropy and Shannon entropy. This shows that controlling the Shannon entropy is not sufficient for some cryptographic tasks.*

a) *Given $\epsilon \in (0,1)$, construct a sequence of random variables ($X_1$, $X_2$, …, $X_n$, …) where $X_n \in \{0,1,\dots,n\}$, such that*

$$\left.\begin{aligned} H(X_n) &\geq (1-\epsilon)\log n \\ H_{\min}(X_n) &= C, \end{aligned}\right\} \forall n \geq N$$

*for some $N \in \mathbb{N}$ and some constant $C > 0$.*

b) *Given $\epsilon \in (0,1)$, construct a sequence of random variables $(X_1, Y_1)$, $(X_2, Y_2)$, …, where $X_n, Y_n \in \{0,1,\dots,n\}$, such that*

$$\left.\begin{aligned} H(X_n) = H_{\min}(X_n) &= \log n && \forall n \\ H(X_n|Y_n) &\geq (1-\epsilon)\log n \\ H_{\min}(X_n|Y_n) &= C \end{aligned}\right\} \forall n \geq N$$

*for some $N \in \mathbb{N}$ and some constant $C > 0$.*

**Exercise 4.2** (Rényi entropy)**.** *Both the min-entropy and the Shannon entropy are limiting cases of the following family of Rényi entropies:*

$$H_\alpha(X) = \frac{1}{1-\alpha}\log\sum_x P(x)^\alpha, \qquad \alpha \in (0,1)\cup(1,+\infty)\,.$$

a) *Compute the limit of the above quantities for $\alpha \to \{0,1,+\infty\}$.*

b) *Plot the Rényi entropy as a function of $\alpha$ for the random variable $X$ distributed as*

| $x$ | 0 | 1 | 2 |
|---|---|---|---|
| $P(x)$ | 1/2 | 1/4 | 1/4 |

c) *Show that, for any random variable $X \in \mathcal{X}$ and any pmf $P(x)$, the Rényi entropy is monotonically non-increasing in the parameter $\alpha$. This yields an alternative proof of the fact that $H_{\min}(X) \leq H(X) \leq \log|\mathcal{X}|$.*

**Hint:** Compute $\frac{\mathrm{d}}{\mathrm{d}\alpha}H_\alpha$ and then express it in the form of $f(\mathbb{E}[t]) - \mathbb{E}[f(t)]$, where $f(t) = t\log t$ and $t = P(x)^{\alpha-1}$.

**Exercise 4.3.** *Consider a source $(X,Y)$ where $Y$ is a uniformly distributed bit and $X$ follows the following conditional distribution:*

| $P_{X\|Y}$ | $x=1$ | $x=2$ | $x=3$ | $x=4$ | $x=5$ |
|---|---|---|---|---|---|
| $y=1$ | $\frac{1}{4}$ | 0 | $\frac{1}{4}$ | $\frac{1}{2}$ | 0 |
| $y=2$ | $\frac{1}{8}$ | $\frac{1}{4}$ | $\frac{1}{8}$ | 0 | $\frac{1}{2}$ |

a) *Find an extractor, a function $f : x \mapsto f(x) \in \{0,1\}$, such that $Z = f(X)$ is uniformly distributed and independent of $Y$.*

b) *Compute the marginal distribution $P_X$, as well the entropy $H_{\min}(X)$.*

c) *Compute the conditional entropy $H_{\min}(X|Y)$.*

d) *Compare $H_{\min}(X)$ with $H_{\min}(X|Y)$.*

- *Why is this relationship between the two quantities expected?*
- *Is the extractor in the first item optimal?*
- *Could there be a better extractor if we only require $Z$ to be uniformly distributed, but not necessarily independent of $Y$?*

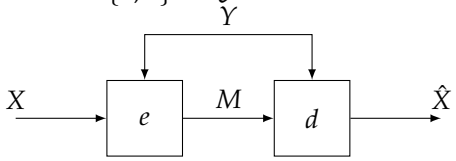
**Exercise 4.4.** *Consider a source producing random variables $X$ and $Y$ with the following joint distribution:*

| $P_{XY}$ | $x=1$ | $x=2$ | $x=3$ | $x=4$ | $x=5$ |
|---|---|---|---|---|---|
| $y=1$ | $\frac{1}{32}$ | $\frac{1}{8}$ | $\frac{1}{16}$ | $\frac{1}{4}$ | $\frac{1}{32}$ |
| $y=2$ | $\frac{1}{8}$ | $0$ | $\frac{1}{8}$ | $\frac{1}{4}$ | $0$ |

a) *Compute $H_{\min}(X|Y)$ and $H(X|Y)$.*

b) *Assume now that $Y=y$ has been observed. Find an optimal source code for $P_{X|Y=y}$ for $y \in \{1,2\}$.*

c) *We are interested in compressing $X$ losslessly. Assume that both the encoder and the decoder have access to $Y$ (see the diagram below). Find a variable-length source code for this setting, i.e., an encoder $e : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^*$ and a decoder $d : \{0,1\}^* \times \mathcal{Y} \to \mathcal{X}$.*




d) *What is the expected codeword length of this code? (Note that you need to take the expectation over both $Y$ and $X$.)*

e) *Find an extractor that extracts exactly $H_{\min}(X|Y)$ bits of uniform and independent randomness from $X$.*

**Exercise 4.5** (AEP for smooth min-entropy)**.** *We are interested in showing that $\frac{1}{n}H^{\epsilon}_{min}(X^n) \to H(X)$ as $n \to \infty$ and $\epsilon \to 0$ (in that order), where $X^n$ is i.i.d. following $P_X$.*

a) *For each $\epsilon > 0$, let $\mathcal{A}^{(n)}_{\epsilon}(X)$ denote the $\epsilon$-typical set of $X$. For each positive integer $n$, define a distribution on $\mathcal{X}^n$ as*

$$\hat{P}_{X^n}(x^n) := \begin{cases} P_{X^n}(x^n) & x^n \in \mathcal{A}^{(n)}_{\epsilon}(X) \\ \frac{1-P_{X^n}(\mathcal{A}^{(n)}_{\epsilon}(X))}{|\mathcal{X}|^n - |\mathcal{A}^{(n)}_{\epsilon}(X)|} & x^n \notin \mathcal{A}^{(n)}_{\epsilon}(X) \end{cases}$$

*Prove that for $n$ large enough, $\delta_{\text{tvd}}(P_{X^n}, \hat{P}_{X^n}) \leq \epsilon$.*

b) *Using $\hat{P}_{X^n}$ from the previous step, show that*

$$\frac{1}{n} H^{\epsilon}_{min}(X^n) \geq H(X) - \epsilon$$

*for $n$ large enough.*

c) *Using the continuity bound from Exercise 3.5, show that*

$$\frac{1}{n} H^{\epsilon}_{min}(X^n) \leq H(X) + \frac{1}{n} h(\epsilon) + \epsilon \cdot \log |\mathcal{X}|$$

*for $0 < \epsilon < 1/2$.*

*The desired limits now follow.*

# 5
# *Error correction codes*

- You are familiar with Hamming distance and can compute code rate and distance.
- You can compute the Hamming bound and check if a code is perfect.
- You can construct a linear code from its generator matrix or parity check matrix.
- You can construct basic Reed-Solomon codes.

## *5.1 Problem setup and definitions*

Error correcting codes are a very rich topic and are studied by communication engineers, computer scientists and mathematicians alike. In computer science, for example, they are used in complexity theory, cryptography, and the study of pseudo-randomness. We can only touch the very surface of this theory here. We will first discuss some general properties of codes and particularly linear codes, and then move on to describe one widely used class of codes in more detail, the Reed-Solomon family of codes.

In this section we discuss codes with a lot of structure. The rich structure is useful in constructing codes that can be efficiently encoded and decoded and in exploring properties like the code distance rigorously. When we move on to channel coding in Chapter 7 we will see that if we just want to argue about the existence of codes then it is often easier to work with random codes, where each codeword is chosen randomly according to some distribution. These two perspectives are quite orthogonal, and in fact the research area dealing with error correcting codes (coding theory) and with random codes (information theory) are distinct and often use different tools.

### *5.1.1 Basic properties of codes*

In the following we will consider codewords that are strings of a fixed length of symbols in some alphabet $\Sigma$. Later on we will assume that $\Sigma$ is a finite field, but for now we just think of it without loss

of generality as the set $\{0, 1, \ldots, |\Sigma| - 1\}$. The following notions are useful.

The *Hamming weight* of a string $x^n \in \Sigma^n$ is defined as

$$w(x^n) := \big|\{i : x_i \neq 0\}\big|, \tag{5.1}$$

i.e., the number of nonzero elements of $x^n$. The *Hamming distance* between two strings $x^n, y^n \in \Sigma^n$ is defined as

$$\delta(x^n, y^n) = \big|\{i : x_i \neq y_i\}\big|, \tag{5.2}$$

i.e., the number of locations where the strings differ.

We will now introduce the notion of an *error correction code*, which we will simply call a code for the remainder of this chapter.

An error correction code of (block) *length* $n$ over a finite alphabet $\Sigma$ is a subset of $\Sigma^n$. It is determined by a set $C \subseteq \Sigma^n$, the *codebook*. The elements of $C$ are called *codewords*.

We will use the following properties and definitions:

- An error correction code is a *binary code* if $\Sigma = \{0, 1\}$.
- The size of the codebook is denoted by $|C|$.
- The *rate* of the code is defined as

$$R(C) = \frac{\log |C|}{\log |\Sigma^n|} = \frac{\log |C|}{n \log |\Sigma|} \tag{5.3}$$

- The minimal *distance* of a code $C$, denoted $d(C)$, is defined as

$$d(C) = \min_{\substack{c, c' \in C \\ c \neq c'}} \delta(c, c') \tag{5.4}$$

On the one hand, the rate of a code tells us how efficient it is. That is, if the rate reaches its maximum, 1, this means that every element of $\Sigma^n$ is a codeword and there is in fact no redundancy in the codewords. If the rate is lower there is more redundancy as only a fraction of the possible strings are used as codewords. On the other hand, the minimum distance tells us how robust the code is—if the distance is large this means that many errors have to occur until one codeword is mistaken for another. Clearly these two parameters cannot be chosen arbitrarily. If we want a larger distance then we generally have to reduce the rate, and vice versa.

### 5.1.2 *Perfect codes*

The following bounds establish a relationship between the code parameters, limiting the size of the codebook in terms of the distance $d$ and the length $n$. It establishes our first fundamental limit.

**Lemma 5.1** (Hamming bound)**.** *Let $C$ be a binary code of length $n$ and distance $d(C) = d$. Then,*

$$|C| \leq 2^n \left( \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} \right)^{-1} \tag{5.5}$$

*In particular, when $d = 3$ we have $|C| \leq \frac{2^n}{n+1}$.*

*Proof.* For every codeword $c$ define its neighbourhood $N(c, r)$ as all the strings that differ from $c$ in at most $r$ locations. Setting $r = \lfloor \frac{d-1}{2} \rfloor$, we note that $N(c, r) \cap N(c', r) = \emptyset$ for any two distinct codewords $c$ and $c'$. Moreover, we have

$$|N(c, r)| = \sum_{i=0}^{r} \binom{n}{i}. \tag{5.6}$$

Hence, we can write

$$2^n \geq \left| \bigcup_{c \in C} N(c, r) \right| = \sum_{c \in C} |N(c, r)| = |C| \sum_{i=0}^{r} \binom{n}{i}. \tag{5.7}$$

Solving this for $|C|$ yields the desired inequality. □

Note that for this bound to hold with equality the space must be exactly filled out by the neighbourhood balls. Codes for which this is true, and the Hamming bound is saturated, we call *perfect codes*. We can also express this result in terms of the code rate. For any binary code $C$, we have

$$R(C) \leq 1 - \frac{1}{n} \log \left( \sum_{i=0}^{\lfloor \frac{d(C)-1}{2} \rfloor} \binom{n}{i} \right). \tag{5.8}$$

### 5.1.3 *Decoding*

The idea of a code is obviously to protect the encoded data from bit flips, erasures, losses, and other more general forms of noise. But how can we recover the original codeword after such noise has been applied? It turns out that really depends on the type of noise. In Chapter 7 we will construct decoders for general probabilistic noise channels. But here, we will instead operate under simple assumptions:

- We assume that every codeword is chosen with equal probability.
- We only consider errors that change or erase symbols in the $n$ bit string, where by erase we mean that a symbol is replaced with a fixed symbol indicating that an error has occured at this location.
- We assume that the probability of seeing an error pattern with errors or erasures at exactly $k$ locations of the string, say $p_k$, is independent of the actual pattern of errors.
- We assume that $p_{k+1} < p_k$, for any $k \geq 0$. In other words, seeing fewer errors is always more likely than seeing more.

This is satisfied for some important channels like the binary symmetric channel or the binary erasure channel.

In this case *minimum distance decoding* is optimal. Say that $\tilde{c}$ is the string we observe after noise has been applied to the codeword. Then, the most likely codeword $\hat{c}$, which is not necessarily unique, satisfies

$$\hat{c} \in \operatorname*{argmin}_{c \in C} \delta(c, \tilde{c}) \tag{5.9}$$

Thus, we simply look for codewords that are closest to the observed string $\tilde{c}$ in Hamming distance. For a large code finding these codewords can become prohibitively expensive, so we are not satisfied with this; but for now it will have to do.

The following relationships between minimal distance of a binary code and its use for error correction are rather immediate. Consider a binary code with minimum distance $d$. Such a code can be used to

- Detect up to $d - 1$ bit flip errors.
- Correct up to $\lfloor \frac{d-1}{2} \rfloor$ bit flip errors.
- Correct up to $d - 1$ erasures.

We see that detecting errors is much easier than correcting them. The simplest example of this is the repetition code that repeats every symbol twice. It can detect 1 error but it cannot correct any. In the erasure model the decoder is informed which bits of the codeword are faulty, which makes the task easier again.

What is the distance of this code?

## 5.2 *Linear codes*

Linear codes are particularly structured as they allow us to see $\Sigma^n$ as a vector space and the codebook as a subspace.

Consider the code $C \in \{0,1\}^n$ where each codeword is constructed by adding a parity bit to a bit string of length $n - 1$. Is this a linear code? What can you say about its minimum distance?

### 5.2.1 *Definition and basic properties*

We consider the vector space $F_q^n$ (of vectors of length $n$ with elements in $F_q$). As usual, addition of vectors is defined as element-wise addition and multiplication with a scalar in $F_q$ is also performed element-wise. On top of that, we have an inner product $\langle a, b\rangle := \sum_{i=1}^n a_i b_i$.

A code is a *linear code* if $\Sigma = F_q$ is a field and if $C$ is a subspace of the vector space $F_q^n$. This means that for any two codewords $c, c' \in C$ and $a \in F_q$, we have that $ac \in C$ and $c + c' \in C$. In particular, the all zero vector is in $C$.

Note that vector addition is the usual element-wise addition in $F_q$ and multiplication is element-wise multiplication with the same scalar in $F_q$. Also take care to distinguish $F_{p^n}$ from $F_p^n$. The first is a finite field itself whereas the latter is a vector space.[1]

[1] One consequence of this is, for example, that multiplication of two elements in $F_q^n$ is not defined.

We first note that the minimal distance of a code can be computed more easily for linear codes.

**Lemma 5.2.** *If $C$ is a linear code, then $d(C) = \min_{c\in C} w(c)$.*

*Proof.* The equality follows from the following observations. For any two codewords $c, c'$, we have that $\delta(c, c') = \delta(c - c', 0) = w(c - c')$. Moreover, since the code is linear, $c'' = c - c'$ is itself a codeword and every codeword can be written in this way. Hence,

$$d(C) = \min_{c,c'} \delta(c, c') = \min_{c,c'} w(c - c') = \min_{c''} w(c'') \,. \tag{5.10}$$

Hence, the minimum over $c, c'$ of the Hamming distance is equal to the minimum Hamming weight of any element of the code. □

Since linear codes form subspaces we can express every codeword as a linear combination of a basis of codewords. We denote by $k$ the *dimension* of the subspace, or the minimal number of codewords needed to form a basis. Let us call this basis $\{b_i\}$ for $i \in \{1, 2, \ldots, k\}$, and note that $b_i \in C$ are themselves codewords. Then every other codeword $c$ can be written as $c = \sum_{i=1}^k \alpha_i e_i$ for some coefficients $\alpha_i \in F_q$. Clearly the codebook size then satisfies $|C| = q^k$.

A linear code with a $k$-dimensional subspace of an $n$-dimensional space is referred to as a $[n, k]_q$*-code*. Furthermore, if it has minimum distance $d$, we call it an $[n, k, d]_q$*-code*.

We usually drop the subscript $q$ when it is clear from context, e.g., when we are discussing binary codes.

The following bound on the codebook size applies to linear codes.

**Lemma 5.3** (Singleton bound)**.** *Let $C$ be a $[n,k,d]_q$-code. Then,*

$$d \leq n-k+1\,, \qquad \textit{or, equivalently,} \qquad |C| \leq q^{n-d+1}\,. \tag{5.11}$$

*Proof.* Let $C$ be an arbitrary code of minimum distance $d$. Clearly, all codewords $c \in C$ are distinct. Moreover, if we puncture the code by deleting the first $d-1$ letters of each codeword, then all resulting codewords must still be pairwise different (because it can correct $d-1$ erasures). The newly obtained codewords each have length $n-(d-1) = n-d+1$, and, thus, there can be at most $q^{n-d+1}$ of them. □

### 5.2.2 *Generator and parity-check matrices*

Linear codes allow for a simple parametrisation in terms of either one of two matrices, the generator and parity-check matrix.

Let $C$ be an $[n,k]_q$-code. A matrix $G \in F_q^{n\times k}$ is said to be a *generator matrix* for $C$ if its $k$ columns span $C$.

Using the generator matrix we can encode any string $x \in F_q^k$ into a codeword $c \in F_q^n$ by the matrix multiplication $c = Gx$. Note that a linear code admits different generator matrices, corresponding to the different choices of basis for the code as a vector space. In practice, this corresponds to different encodings of the messages into codewords, with the same fixed set of codewords.

There are two generic ways two characterise a subspace:

- By specifying a basis of the subspace $C$, as we have done above using the generator matrix.
- By specifying a basis of the orthogonal subspace, $C^\perp$.

For linear codes that orthogonal subspace is spanned by vectors that are orthogonal to the linear subspace spanned by the codewords. Those vectors can be interpreted as parity checks.

Let $C$ be an $[n,k]_q$-code. A matrix $H \in F_q^{(n-k)\times n}$ is said to be a *parity check matrix* for $C$ if its rows span $C^\perp$.

Thus, in particular, the paritiy check matrix must satisfy $Hc = 0$ for every $c \in C$, or, equivalently, $HG = 0$.

**Example.** *The Hamming code is a binary $[7,4,3]$-code given by codewords of the form*

$$x_1,\quad x_2,\quad x_3,\quad x_4,\quad x_2 \oplus x_3 \oplus x_4,\quad x_1 \oplus x_3 \oplus x_4,\quad x_1 \oplus x_2 \oplus x_4\,. \tag{5.12}$$

**Example.** *Consider the binary repetition code for $n = 3$ comprised of the codewords* 000 *and* 111*. The generator matrix for this code is $G = (1,1,1)^T$.*

*A possible generator matrix for this code is given by*

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix} \tag{5.13}$$

*A possible parity check matrix is given by*

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \tag{5.14}$$

*The Hamming code is a perfect code since* $2^k = \frac{2^n}{n+1}$ *for* $k = 4$ *and* $n = 7$.

### 5.2.3 *Dual code*

Every linear has its dual, since every subspace has its orthogonal complement.

The dual of a $[n,k]_q$-code $C$, the $[n, n-k]_q$-code $C^\perp$. That is, the space spanned by all codewords $c' \in F_q^n$ such that

$$\sum_{i=1}^{n} c_i c_i' = 0 \tag{5.15}$$

for all $c \in C$.

From the definition we can see that $G^\perp = H^T$ and $H^\perp = G^T$. In particular, the dual of a dual code is the code itself.

### 5.2.4 *Decoding*

Decoding becomes a bit easier for linear codes. We can write $\tilde{c} = c + e$ where $c$ is the true codeword and $e$ is the error. We first compute

$$H\tilde{c} = Hc + He = He, \tag{5.16}$$

where we used that $Hc = 0$ for every codeword. The goal is then to find an $e$ with minimal Hamming weight such that $He = H\tilde{c}$, i.e.,

$$\hat{e} \in \operatorname*{argmin}_{e:He=H\tilde{c}} w(e), \qquad \text{and,} \qquad \hat{c} = \tilde{c} - \hat{e}\,. \tag{5.17}$$

If the code is not too large we can in fact build a table that maps each of the $q^{n-k}$ possible values of $H\tilde{c}$ to a corresponding error string $\hat{e}$.

Decoding can then be done by looking up $H\tilde{c}$ in the table, revealing $\hat{e}$, and computing $\hat{c} = \tilde{c} - \hat{e}$.

Note that for large codes this is still infeasible as the lookup table has exponentially (in $n$) many elements. This motivates the study of linear codes with additional structure, in the hope that they will make decoding more efficient.

## 5.3 *Reed-Solomon codes*

Reed-Solomon were first used to do error correction for the Voyager program and became really widespread in their use to protect against errors on compact discs. They are still used in two-dimensional bar codes like QR codes.

The Reed-Solomon code is actually a family of codes, where every code is characterised by three parameters: an alphabet size $q$, a block length $n$, and a message length $k$, with $k < n \leq q$. In this code a message $m = (m_0, m_1, \ldots, m_{k-1}) \in F_q^k$ is first mapped to a polynomial $p_m(x)$ with $x \in F_q$ of degree $k-1$ given by

$$p_m(x) = \sum_{i=0}^{k-1} m_i x^i. \tag{5.18}$$

The codeword for $m$ is then obtained by evaluating $p_m$ at $n$ different points $x_i \in F_q$ for $i \in [n]$, i.e.

$$c(m) = (p_m(x_1), \ldots, p_m(x_n))^T. \tag{5.19}$$

This constitutes a linear code with the generator matrix $G \in F_q^{n \times k}$ given by

$$G = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_2 & \ldots & x_n \\ \vdots & \vdots & & \vdots \\ x_1^{k-1} & x_2^{k-1} & \ldots & x_n^{k-1} \end{pmatrix}^T \tag{5.20}$$

The basic idea is that a polynomial of order $k-1$ is uniquely specified if we know its value at $k$ points or more.

**Lemma 5.4.** *The Reed-Solomon code is a $[n, k, n-k+1]_q$-code.*

In particular, Reed-Solomon codes satisfy the singleton bound with equality.

*Proof.* The code is an $[n, k]_q$-code by construction. The only property we need to show here is that the minimal distance of the code satisfies $d \geq n - k + 1$.

By Lemma 5.2, the minimal distance of the codebook is simply given by the minimal Hamming weight of its nonzero codewords. But since for any $m \neq 0$, the polynomial $p_m(x)$ is nontrivial and of order at most $k-1$, we know that it has at most $k-1$ roots. Thus, the number of zeros in the codeword is at most $k-1$. Hence, we must have $d \geq n-(k-1) = n-k+1$.

On the other hand, we observe that by the Singleton bound we must have $d \leq n-k+1$, therefore concluding the proof. □

**Example.** *If we choose $q = 2^n$ we can interpret the Reed-Solomon code as a binary code. For $q = 2^2$, $n = 4$ and $k = 2$ we get the following mappings, where each symbol {0, 1, 2, 3} can be interpreted as a binary sequence $\{00, 01, 10, 11\}$. The codewords are constructed by evaluating the polynomial at the points $\{0, 1, 2, 3\}$, using the multiplication and addition rules for $F_4$ discussed in Section 0.4. For example, we get*

$$1100 \sim (3,0) \to 3+0x \to \{3,3,3,3\} \sim 11111111 \tag{5.21}$$

$$0110 \sim (1,2) \to 1+2x \to \{1,3,2,0\} \sim 01111000 \tag{5.22}$$

$$1011 \sim (2,3) \to 2+3x \to \{2,1,3,0\} \sim 10011100\,. \tag{5.23}$$

What is the encoding of the strings 0000 and 0110?

*The above should be read as "initial bit string" $\sim$ "written as two values $(m_0, m_1)$ in $F_4$ by interpreting it as a binary representation" $\to$ "corresponding polynomial of degree 1" $\to$ "polynomial evaluated at $x = \{0,1,2,3\}$ $\sim$ "encoded bit string using binary representation".*

## 5.4 Exercises

**Exercise 5.1.** *Consider a code that stores $k = 4$ bits $x_1, x_2, x_3$ and $x_4$ in $n = 8$ bit codewords by computing the parities $x_1 \oplus x_2$, $x_3 \oplus x_4$, $x_1 \oplus x_3$ and $x_2 \oplus x_4$ and storing them together with the original bits.*

a) *Give the codewords for this code and compute the minimal distance. How many errors can it detect and correct?*

b) *Is it a linear code? If so, compute matrices G and H.*

c) *Use the Hamming bound to determine if this code is perfect or not.*

d) *Construct the dual code for this code.*

**Exercise 5.2.** *Explicitly construct the dual code of the Hamming code (that is, without using the relations $G^\perp = H^T$ and $H^\perp = G^T$). What is its rate and code distance?*

**Exercise 5.3.** *Show the following properties of a (binary) linear $[n,k]$-code C.*

a) *If H is the parity check matrix of C, then $d(C)$ equals the minimal number of columns of H that are linearly dependent.*

b) *Prove that (after permuting the coordinates if necessary) C has a generator matrix of the form* $G = [I_k \; G']^T$ *where* $I_k$ *is the* $k \times k$ *identity matrix, and where* $G'$ *is some* $k \times (n-k)$ *matrix.*

**Exercise 5.4.** *Assume you are given a linear code by its* $n \times k$ *generator matrix* $G$ *or by its* $(n-k) \times n$ *parity-check matrix* $H$*. Which of the below operations can cause a reduction or increase in the minimum code distance?*

a) *Exchanging two rows of* $G$*.*

b) *Exchanging two rows of* $H$*.*

c) *Exchanging two columns of* $G$*.*

d) *Exchanging two columns of* $H$*.*

e) *Deleting a row of* $G$*.*

f) *Deleting a row of* $H$*.*

g) *Deleting a column of* $G$*.*

h) *Deleting a column of* $H$*.*

i) *Adding one column of* $H$ *to another column of* $H$*.*

**Exercise 5.5.** *Consider the field* $F_{2^n}$*. For every* $s \in F_{2^n}$ *with* $s \neq 0$*, define the function*

$$f_s : \{0,1\}^n \to \{0,1\}^\ell, \quad x \mapsto x \cdot s \mod 2^\ell,$$

*where the multiplication is with regards to the field* $F_{2^n}$*. Show that the family* $\{f_s\}_s$ *is a two-universal family of hash functions.*

# 6
# Communication Channels

**Intended learning outcomes:**

- You know the BSC, BEC and AWGN channels and can compute their channel mutual information.
- You can compute the channel mutual information for other simple discrete channels and know how to simplify the calculation in the presence of symmetries.
- You understand the geometric interpretation of the channel mutual information as a relative entropy radius of the channel image.
- You can compute differential entropy and mutual information for Gaussian distributions.

**Book reference:** Sections 7.1–7.3 in Cover & Thomas[1].

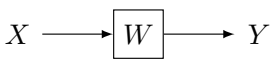
[1] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991. ISBN 9780471748823. DOI: 10.1002/047174882X

## 6.1 Point-to-point channels

We have encountered discrete channels previously, for example when we discussed data-processing inequalities, and seen that they simply correspond to conditional pmfs. Let us still fix some nomenclature.

A *channel* $W$ from $\mathcal{X}$ to $\mathcal{Y}$ is a conditional probability distribution $W_{Y|X}(y|x)$, where $x \in \mathcal{X}$ is a *channel input* symbol and $y \in \mathcal{Y}$ is a *channel output* symbol. The sets $\mathcal{X}$ and $\mathcal{Y}$ are called channel *input alphabet* and channel *output alphabet*, respectively.

The random variable $X$ is the channel input, and its distribution is not a property of the channel. However, once it is fixed, the channel will induce a joint distribution on $X$ and the channel output $Y$.




The input and output alphabets can be either discrete or continuous. If the output alphabet is discrete then $W(\cdot|x)$ is a pmf and if the output alphabet is continuous we will use the lower-case notation

$w(\cdot|x)$ to make clear that it is a pdf. The conditional probabilities simply express with what likelihood a certain output symbol $y$ appears given that the channel input is set to symbol $x$.

For discrete alphabets a graphical representation is also useful, where we draw the conditional probabilities on a graph connecting the input symbols with output symbols. We start with some important examples of discrete channels:

1. The *binary symmetric channel* (BSC) takes a binary input to a binary output. The bit is flipped with a certain probability, here denoted $\epsilon$, and otherwise left intact:

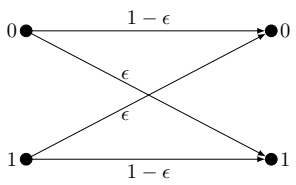



Figure 6.1: Binary symmetric channel with cross-over probability $\epsilon$.

The conditional probability distribution of the channel is given by

$$W_{\mathrm{BSC}(\epsilon)}(y|x) = (1-\epsilon)\mathbf{1}\{x = y\} + \epsilon\,\mathbf{1}\{x \neq y\}\,. \tag{6.1}$$

2. The *binary erasure channel* (BEC) takes a binary input to a ternary output, $\{0, 1, \perp\}$. The output $\perp$ has probability $\epsilon$ on either input, and otherwise the input symbol remains unaffected. Essentially this is a channel that flags errors:

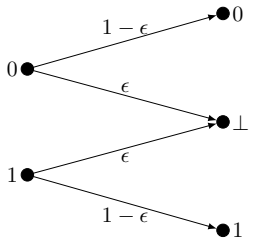



Figure 6.2: Binary erasure channel with error probability $\epsilon$.

The conditional probability distribution of the channel is given by

$$W_{\mathrm{BEC}(\epsilon)}(y|x) = (1-\epsilon)\mathbf{1}\{x = y\} + \epsilon\,\mathbf{1}\{y = \perp\}\,. \tag{6.2}$$

We will be able to treat general discrete channels in a unified matter in the following, but the assumption that the input and output alphabets are finite will be crucial. However, some channels with continuous input and output are also of great practical relevance. We will restrict our attention to a certain class of Gaussian channels.

3. The *additive white Gaussian noise (AWGN)* channel takes an input $X \in \mathbb{R}$ and outputs

$$Y = X + Z, \tag{6.3}$$

where $Z$ follows a Gaussian distribution with mean 0 and standard deviation $\sigma$, and is independent of $X$. The channel behaviour can thus be characterised by the conditional pdf

$$w_{\mathrm{AWGN}(\sigma)}(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x)^2}{2\sigma^2}}, \tag{6.4}$$

which is a Gaussian pdf with mean $x$ and standard deviation $\sigma$.

## 6.2 Channel information for discrete channels

For now we restrict ourselves to discrete channels and explore the following quantity, the channel mutual information. It measures the maximal correlation (measured in mutual information) between the input and output random variables that a channel can support. In the following we will introduce its definition and explore some of its basic properties.

### 6.2.1 Definition and basic properties

Let $\mathcal{X}$ and $\mathcal{Y}$ be discrete alphabets and let $W$ be a channel from $\mathcal{X}$ to $\mathcal{Y}$. The *channel (mutual) information* of $W$ is defined as

$$I(W) := \max_{P_X} I(X:Y)\,, \tag{6.5}$$

where the mutual information is evaluated for

$$P_{XY}(x,y) = P_X(x) W_{Y|X}(y|x), \tag{6.6}$$

is the joint distribution of channel input and output, and $P_X$ is optimised over all pmfs on the channel input alphabet $\mathcal{X}$.

This is the maximal mutual information between channel input and output. The quantity is often called "channel capacity" in the literature, and we will see that it in fact corresponds to the maximal rate at which information can be transmitted over the channel in Chapter 7. However, we prefer to keep a semantic difference between information quantities, like the channel information, and operational quantities, like the channel capacity. Only through the study of information theory do we actually establish their equivalence, and usually only in special cases, e.g. for discrete memoryless channels in this case.

The optimisation is well-behaved since the underlying function is concave in $P_X$, as the following lemma shows.

**Lemma 6.1.** *For a fixed channel $W$, the mutual information between channel input and output, $I(X:Y)$, is concave in the pmf $P_X$.*

*Proof.* We have $I(X:Y) = H(Y) - H(Y|X)$, which we may write as

$$I(X:Y) = H(Y) - \sum_x P_X(x) H(Y|X=x), \tag{6.7}$$

where $P_Y(y) = \sum_x P_X(x) P_{Y|X}(y|x)$. By concavity of the entropy function we now see that the first term is concave in $P_X$. The second term is linear and thus concave in $P_X$ as well. □

This property is what allows us to efficiently numerically compute the channel mutual information if we must. More precisely, concavity ensures that if we find a local maximum for the mutual information then that maximum is in fact global. Based on this, there are algorithms that can compute the channel mutual information efficiently for any stochastic map. However, in this course we will focus on examples where we can do the optimisation by hand.

Some further properties follow immediately from the corresponding properties of the mutual information. For example, we have

$$0 \leq I(W) \leq \min\left\{ \log|\mathcal{X}|, \log|\mathcal{Y}| \right\}. \tag{6.8}$$

Show this using the bounds discussed in Chapter 1.

### 6.2.2 *Evaluation for symmetric channels*

One very important consequence of this concavity is that we can simplify the optimisation for symmetric channels. The strongest symmetry we consider here is one where every permutation of input symbols can be "undone" be doing a respective permutation of the output symbols of the channel.

The permutation group $S_{\mathcal{X}}$ contains all bijective functions from $\mathcal{X} \to \mathcal{X}$ and each $\pi \in S_{\mathcal{X}}$ corresponds to a relabelling of the different input symbols in $\mathcal{X}$. The total number of permutations is given by $|S_{\mathcal{X}}| = |\mathcal{X}|!$.

We show a therem here that applies already if the channels has limited symmetry, namely if only some subgroup of input permutations can be "undone" at the channel output.

**Proposition 6.2.** *Consider a channel $W$ such that for some subgroup of permutation $S \subseteq S_{\mathcal{X}}$ and every $\pi \in S$, there exists a permutation $\widetilde{\pi} \in S_{\mathcal{Y}}$ such that $W(y|x) = W(\widetilde{\pi}(y)|\pi(x))$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. Then, there is an input distribution $Q_X$ achieving the channel mutual information that satisfies $Q_X(x) = Q_X(\pi(x))$ for all $\pi \in S$ and $x \in \mathcal{X}$.*

Notably, when $S$ is the cyclic group on $\mathcal{X}$ (or if $S = S_{\mathcal{X}}$), then the only input distribution that satisfies the above property is the uniform distribution, $P_X(x) = \frac{1}{|\mathcal{X}|}$. In this case the channel information

is simply given by $I(X:Y)$ where $X$ is uniformly distributed, and no maximisation is needed.

*Proof.* Let $d = |\mathcal{X}|$. Assume $P_X$ is a pmf that achieves the channel mutual information. In a first step, we want to show that $I(X:Y)_P = I(X:Y)_{P^\pi}$ for any permutation $\pi \in S$ and $P_X^\pi(x) = P_X(\pi^{-1}(x))$. For this purpose we introduce the random variable $\pi(X)$ and note that

$$X \longleftrightarrow \pi(X) \longleftrightarrow \widetilde{\pi}(Y) \longleftrightarrow Y \tag{6.9}$$

form a Markov chain. Hence, by data-processing $I(X:Y)_{P^\pi} \geq I(X:Y)_P$ —but since we started with the assumption that $P_X$ is a maximiser the two mutual informations must in fact be equal.

Next we use this identity to write

$$I(W) = I(X:Y)_P = \sum_{\pi\in S} \frac{1}{|S|} I(X:Y)_{P^\pi}\,. \tag{6.10}$$

This can be interpreted as the expectation value of $I(X:Y)$, where the pmf is chosen uniformly at random from amongst the permuted pmfs $P^\pi$. However, since the mutual information is concave in the pmf, we have that

$$\sum_{\pi\in S} \frac{1}{|S|} I(X:Y)_{P_k} \leq I(X:Y)_Q, \tag{6.11}$$

where

$$Q_X(x) = \sum_{\pi\in S} \frac{1}{|S|} P_X^\pi(x) = \sum_{\pi\in S} \frac{1}{|S|} P_X(\pi^{-1}(x)) \tag{6.12}$$

is the expected pmf of $X$. Hence, the pmf $Q_X$ performs at least as good as $P_X$, and thus must also achieve the channel information. Finally, we show that $Q_X(\pi(x)) = Q_X(x)$. To see this, we use that $S$ is a group and

$$Q_X(\pi(x)) = \sum_{\bar{\pi}\in S} \frac{1}{|S|} P_X(\bar{\pi}^{-1}\cdot\pi(x)) \tag{6.13}$$

$$= \sum_{\bar{\pi}\in S} \frac{1}{|S|} P_X(\bar{\pi}^{-1}(x)) = Q_X(x) \tag{6.14}$$

by a change of variable. □

### 6.2.3 *Examples: BSC and BEC*

We will now consider two very prominent examples of communication channels and compute their channel information.

1. Binary Symmetric Channel: Recall the BSC from Figure 6.1. The channel mutual information for the BSC is easy to evaluate, even

without invoking Proposition 6.2 — which does clearly apply here. Let us simply note that $H(Y|X = x) = h(\epsilon)$, the binary entropy evaluated for $\epsilon$, and this is independent of $x \in \{0,1\}$. Hence the mutual information is given by $I(X : Y) = H(Y) - h(\epsilon)$, which is maximised when $Y$ is uniformly distributed. This is achieved when $X$ is uniformly distributed itself. Hence, the channel information is given by

$$I(W_{\mathrm{BSC}(\epsilon)}) = 1 - h(\epsilon)\,. \tag{6.15}$$

2. Binary Erasure Channel: Recall the BEC from Figure 6.2. By Proposition 6.2 we can again argue that the maximising input distribution is the uniform distribution. And we get the output distribution

$$P_Y(y) = \begin{cases} \frac{1}{2}(1-\epsilon) & \text{if } y \in \{0,1\} \\ \epsilon & \text{if } y = \perp \end{cases} \tag{6.16}$$

We again have $H(Y|X = x) = h(\epsilon)$ independent of $x$ and can then compute

$$\begin{aligned} I(W_{\mathrm{BEC}(\epsilon)}) &= H(Y) - h(\epsilon) && (6.17)\\ &= -2 \cdot \frac{1}{2}(1-\epsilon)\log\frac{1}{2}(1-\epsilon) - \epsilon\log\epsilon - h(\epsilon) && (6.18)\\ &= (1-\epsilon) + h(\epsilon) - h(\epsilon) && (6.19)\\ &= 1 - \epsilon && (6.20) \end{aligned}$$

Further examples are discussed in the exercises. However, note that it is not always easy to perform the maximisation over input distributions[2] and thus analytical solutions for the channel information optimisation problem are rare.

[2] Unless the channel has symmetry, as we have seen in the previous section.

### 6.2.4 *Geometric interpretation*

The following expression for the channel information is useful to know, and expresses it as the relative entropy radius of the channel image. This is a geometric measure of how distinct the channel outputs can get for different inputs. We will need it later to prove the converse of the noisy channel coding theorem.

**Proposition 6.3.** *For any channel $W$, we have*

$$I(W) = \min_{Q_Y} \max_{x \in \mathcal{X}} D(W_{Y|X}(\cdot|x) \| Q_Y)\,, \tag{6.21}$$

*where the minimisation is over all pmfs on $\mathcal{Y}$.*

In particular, there exists a pmf $Q_Y^*$ such that

$$D(W_{Y|X}(\cdot|x)\|Q_Y^*) \leq I(W) \qquad \text{for all } x \in \mathcal{X}. \tag{6.22}$$

*Proof.* We first write, using the definition of the channel information,

$$I(W) = \max_{P_X} D(P_{XY}\|P_X \times P_Y) \tag{6.23}$$

$$= \max_{P_X} \min_{Q_Y} D(P_{XY}\|P_X \times Q_Y), \tag{6.24}$$

where the second equality comes from the fact that $D(P_{XY}\|P_X \times Q_Y) = D(P_{XY}\|P_X \times P_Y) + D(P_Y\|Q_Y)$ and the minimum is thus achieved for $Q_Y = P_Y$. If we further rewrite

$$D(P_{XY}\|P_X \times Q_Y) = \sum_{x\in\mathcal{X}} P_X(x) D(W_{Y|X}(\cdot|x)\|Q_Y) \tag{6.25}$$

we realise that this quantity is linear in $P_X$ and convex in $Q_Y$. The idea then is to use Sion's minimax theorem (Prop. 0.13), which states that the minimum and maximum in the above expressions can be interchanged. Hence, we get

$$I(W) = \min_{Q_Y} \max_{P_X} \sum_{x\in\mathcal{X}} P_X(x) D(W_{Y|X}(\cdot|x)\|Q_Y)\,. \tag{6.26}$$

Finally note that the maximum in the above expression is taken for a $P_X$ that is concentrated on a single point. This yields the expression in (6.21). □

## 6.3 Channel information with power constraints

Let us now consider channels with continuous input and/or output alphabets. In fact, let us focus our attention on the most prominent such channel, the AWGN channel. We can ask the usual question about this channel — what is the maximal mutual information that can be reached between channel input and output?

It turns out that without further restrictions the channel mutual information is unbounded. To see this, we simply distribute inputs onto an infinite lattice of values $x_r \in \mathbb{R}$ that are sufficiently separated so that even after the noise is added $w(y|x_r)$ and $w(y|x_{r'})$ only have small overlap for distinct inputs $r$ and $r'$. If the grid distance is chosen to be $6\sigma$, for example, we will get a confusion error that is lower than 0.5%, leading to an almost perfect correlation between channel inputs and outputs. And since we can choose the set of inputs arbitrarily large (as the lattice stretches to infinity), the channel mutual information is unbounded.

Formally construct such an encoder and decoder and compute the probability of error.

In practical applications an AWGN channel for example arises when we encode information in an electromagnetic field, and $X_i$ and $Y_i$ for each channel use $i \in [n]$ are then simply amplitudes of the field. On the other hand, the energy stored in the field grows with the square of the amplitude and needs to be invested by the sender of the electromagnetic pulse. It is natural to restrict how much energy per channel-use, or power, is available at the source.[3] Formally, this is done by requiring that every codeword $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ satisfies

$$\frac{1}{n}\sum_{i=1}^{n} x_i^2 \leq P \tag{6.27}$$

[3] The nomenclature makes sense since power is energy per time unit, and channel uses are temporally separated in this context.

This communication channel models many practical channels, including wireless and satellite links. The noise may be due to a variety of (independent) microscopic reasons; however, the central limit ensures that collectively these noise sources resemble an additive noise with a Gaussian distribution. When we look at the channel mutual information now, we are not interested in optimising over all possible input pdfs, but only those that satisfy the constraint $\mathbb{E}[X^2] \leq P$, i.e., the expected power is bounded by $P$.

The *channel (mutual) information of w with power constraint P* is

$$I(w, P) := \max_{p_X : \mathbb{E}[X^2] \leq P} I(X : Y), \tag{6.28}$$

where we maximise over all pdfs $p_X$ on $X$ such that $\mathbb{E}[X^2] \leq P$.

The goal of this section is to analyse the channel mutual information of the AWGN channel under the above power constraint. We have not discussed continuous variables in any detail and indeed all our definitions of information quantities and our proofs so far have assumed that the random variables take values in a finite alphabet. Hence, before we can fully make sense of the above definition, we will first need to generalise important concepts like entropy and mutual information to the continuous variable setting.

### 6.3.1 *Differential entropy and mutual information*

Let $X$ be a real-valued continuous random variable with support on $\mathcal{S} \subseteq \mathcal{X}$ and pdf $p_X$. The *differential entropy* of $X$ is defined as

$$h(X) = -\int_{\mathcal{S}} p_X(x) \log p_X(x) \mathrm{d}x\,. \tag{6.29}$$

It is worth noting that this integral does not always exist and might in fact be infinite in many cases.

For maths enthusiasts: Construct an example with a valid pdf for which the integral diverges and one for which it becomes negative.

A class of distributions for which it is relatively well-behaved is the uniform distribution, where $p_X(x) = \frac{1}{a}$ in an interval $[0, a]$ and zero elsewhere. In this case it is easy to verify that we have $h(X) = \log a$. An other interesting case is the case of Gaussian distribution with

$$p_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} . \tag{6.30}$$

In this case we can evaluate the differential entropy as follows

$$\begin{aligned} h(X) &= -\int p_X(x) \log p_X(x) \mathrm{d}x && (6.31)\\ &= \int p_X(x) \left( \frac{(x-\mu)^2}{2\sigma^2} \log e + \log \sqrt{2\pi\sigma^2} \right) \mathrm{d}x && (6.32)\\ &= \mathbb{E}[(x-\mu)^2] \frac{\log e}{2\sigma^2} + \frac{1}{2} \log (2\pi\sigma^2) && (6.33)\\ &= \frac{1}{2} \log (2e\pi\sigma^2) . && (6.34) \end{aligned}$$

It is important to note that since the differential entropy can get negative for small $a$ or $\sigma$ it is hard to give it operational meaning.

One thing we can immediately observe is that the differential entropy is independent of the mean of $X$. This is true more generally: $h(X) = h(X + c)$ for any constant $c$, which can be verified by a simple change of variable. Note, however, that $h(cX) = h(X) + \log |c|$, so the entropy is not invariant under rescaling. This can be seen in contrast to the invariance of the entropy of discrete random variables under change of labels (in this case due to rescaling).

We can define conditional entropy and mutual information analogously to the discrete case.

Let $X$ and $Y$ be real-valued continuous random variables with joint pdf $p_{XY}$ with support on $\mathcal{S} \subseteq \mathcal{X} \times \mathcal{Y}$. The *mutual information* between $X$ and $Y$ is defined as

$$I(X : Y) = \int_{\mathcal{S}} p_{XY}(x, y) \log \frac{p_{XY}(x, y)}{p_X(x) p_Y(y)} \mathrm{d}x \, \mathrm{d}y . \tag{6.35}$$

We can again decompose $I(X : Y) = h(X) - h(X|Y)$, assuming that all the expressions are finite, where the conditional differential entropy of $X$ given $Y$ is defined as

$$h(X|Y) = -\int_S p_{XY}(x, y) \log p_{X|Y}(x|y) \, \mathrm{d}x \, \mathrm{d}y , \tag{6.36}$$

and $p_{X|Y}(x|y)$ is the conditional pdf of $X$ given $Y$.

The mutual information for continuous variables is very naturally linked to mutual information for discrete variables. To see this,

consider random variables $X$ and $Y$ supported on $\mathbb{R}$. For every $\Delta > 0$, we then introduce the discrete random variables $X^\Delta$ and $Y^\Delta$ that take values $(r, t)$ for $r, t \in \mathbb{Z}$ with joint pmf

$$P_{X^\Delta Y^\Delta}(r,t) := \int_{x_r-\Delta/2}^{x_r+\Delta/2} \int_{y_t-\Delta/2}^{y_t+\Delta/2} p_{XY}(x,y)\,\mathrm{d}x\mathrm{d}y \tag{6.37}$$

$$\approx \Delta^2 p_{XY}(x_r, y_t), \tag{6.38}$$

where $x_r = r\Delta$ and $y_t = t\Delta$ and the approximation holds for small $\Delta$ as long as the pdf is continuous. The new random variables simply describe discretised versions of $X$ and $Y$, where all probability mass in an interval of length $\Delta$ is course-grained into a single discrete value. Note, however, that $X^\Delta$ and $Y^\Delta$ can take any of countably infinite values, and this thus already extends a bit the definitions we have worked with so far. However, as long as the infinite sums converge, we do not encounter any issues. We may define, for example,

$$I(X^\Delta : Y^\Delta) := \sum_{r=-\infty}^{\infty} \sum_{t=-\infty}^{\infty} P_{X^\Delta Y^\Delta}(r,t) \log \frac{P_{X^\Delta Y^\Delta}(r,t)}{P_{X^\Delta}(r) P_{Y^\Delta}(t)}\,, \tag{6.39}$$

where the marginal pmfs $P_{X^\Delta}$ and $P_{Y^\Delta}$ are defined as usual.

Assume now that $p_{XY}$ is sufficiently well-behaved so that the log-likelihood ratio $\log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)}$ is Riemann-integrable. Then,

$$I(X:Y) = \int_{\mathbb{R}\times\mathbb{R}} p_{XY}(x,y) \log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)}\,\mathrm{d}x\mathrm{d}y \tag{6.40}$$

$$= \lim_{\Delta\to 0} \sum_{r=-\infty}^{\infty} \sum_{t=-\infty}^{\infty} \Delta^2 p_{XY}(x_r, y_t) \log \frac{p_{XY}(x_r, y_t)}{p_X(x_r)p_Y(y_t)} \tag{6.41}$$

$$= \lim_{\Delta\to 0} I(X^\Delta : Y^\Delta). \tag{6.42}$$

An analogous result does not hold for differential entropy. This is evident simply from the fact that fine-graining a random variable will generally strictly increase its entropy, and thus the entropy would always diverge to infinity in such a limit.

Argue that this implies that the mutual information is always non-negative even for continuous variables.

Finally, we can define the relative entropy between two pdfs $p_X$ and $q_X$ as

$$D(p_X \| q_X) = \int_{\mathcal{S}} p_X(x) \log \frac{p_X(x)}{q_X(x)}\,\mathrm{d}x\,, \tag{6.43}$$

where $\mathcal{S} \subseteq \mathcal{X}$ is the support of $q_X$. The same argument we used in Chapter 1, based on Jensen's inequality, reveals that

$$D(p_X \| q_X) \geq 0 \tag{6.44}$$

for all pairs of pdfs.

### 6.3.2 *Channel mutual information of the AWGN channel*

Recall the AWGN channel from (6.4). Its only parameter is the variance $\sigma$, which determines how much noise is added. It turns out that its channel mutual information can be expressed conveniently in terms of the *signal-to-noise ratio*, given by $\frac{P}{\sigma^2}$.

**Theorem 6.4.** *For an AWGN channel $w$ with variance $\sigma^2$ and power constraint $P$, we have*

$$I(w, P) = \frac{1}{2} \log \left(1 + \frac{P}{\sigma^2}\right) . \tag{6.45}$$

*Proof.* We may rewrite $I(X : Y) = h(Y) - h(Y|X)$ where

$$h(Y|X) = h(X + Z|X) = h(Z) = \frac{1}{2} \log(2\pi e \sigma^2) \tag{6.46}$$

can be simplified immediately. We now make the following observations. Using that $Y = X + Z$, we find

$$\begin{aligned} \mathbb{E}[Y^2] &= \mathbb{E}[X^2 + 2XZ + Z^2] && (6.47) \\ &= \mathbb{E}[X^2] + 2\mathbb{E}[X]\mathbb{E}[Z] + \mathbb{E}[Z^2] \leq P + \sigma^2 && (6.48) \end{aligned}$$

since $X$ and $Z$ are independent and $\mathbb{E}[Z] = 0$. So we have a bound on the variance of $Y$ — does this allow us to conclude anything about its differential entropy?

We now argue that $h(Y)_p$ cannot exceed the entropy of a Gaussian $\phi_Y$ with the same variance as $p_Y$. To see this, we first note that we can assume without loss of generality that both $p_Y$ and $\phi_Y$ have mean zero as the entropy is independent of constant shifts. Using this, we can write

$$0 \leq D(p_Y \| \phi_Y) = -h(Y)_p + \int p_Y(y) \log \frac{1}{\phi_Y(y)} \mathrm{d}y . \tag{6.49}$$

Now we note that since $\phi_Y$ is Gaussian the expression $\log \phi_Y(y)$ is of the form $A + By^2$ for two constants $A$ and $B$. Notably, this integral thus only depends on the second moment of $y$. Hence, we may replace $p_Y(y)$ with $\phi_Y(y)$ in (6.49) since they both have the same second moment. Therefore, we have shown that

$$0 \leq -h(Y)_p + h(Y)_\phi \iff h(Y)_p \leq h(Y)_\phi . \tag{6.50}$$

Using this, we can conclude that

$$h(y)_p \leq \frac{1}{2} \log(2\pi e(P + \sigma^2)) , \tag{6.51}$$

and, thus,

$$I(X:Y) \leq \frac{1}{2}\log(2\pi e(P+\sigma^2)) - \frac{1}{2}\log(2\pi e\sigma^2) \tag{6.52}$$

$$= \frac{1}{2}\log\left(1+\frac{P}{\sigma^2}\right) \tag{6.53}$$

Finally, we can verify that equality can be achieved by choosing $p_X$ Gaussian with standard deviation $P$ (and zero mean) as the input distribution. □

## 6.4 Exercises

**Exercise 6.1** (Deterministic channel)**.** *Consider a memoryless channel that takes two bits $(X_1, X_2)$ as input and maps them to two bits of output $(Y_1, Y_2)$ as follows:* $00 \to 01$, $01 \to 10$, $10 \to 11$, $11 \to 00$.

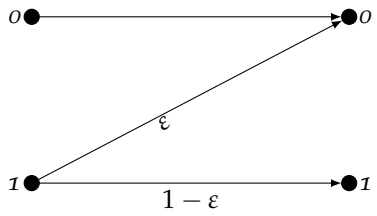
a) *Calculate the mutual information $I(X_1, X_2; Y_1, Y_2)$ for a given joint pmf of the four pairs of input bits. You can express your answer in terms of $p_{ij} = P(X_1 = i, X_2 = j)$ for $i, j \in \{0, 1\}$.*

b) *Show that the channel mutual information is* 2.

c) *Show that, surprisingly, $I(X_1; Y_1) = 0$ for the distribution that achieves the maximal mutual information in part (b) (that is, information is only transferred by considering both bits).*

**Hint:** Find the joint pmf of $X_1$ and $Y_1$.

**Exercise 6.2.** *The Z channel is a binary channel with conditional pmf $p(0|0) = 1$, $p(0|1) = \epsilon$ and $p(1|1) = 1 - \epsilon$.*




a) *Suppose $\epsilon = 1/2$, compute the channel mutual information.*

b) *Plot the channel mutual information as a function of $\epsilon \in [0, 1]$.*

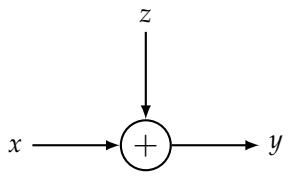
**Hint:** You will need to perform some numerical optimisations here.

**Exercise 6.3.** *For two positive integers k and m, let $k \mod m$ be the* remainder *when k is divided by m. Find the channel mutual information of the m-input discrete memoryless channel in which*

$$Y = X + Z \mod m,$$

*where $X \in \{0, 1, \ldots, m-1\}$, $P[Z = 1] = \frac{3}{4}$, and $P[Z = 0] = \frac{1}{4}$.*

**Exercise 6.4.** *Find the channel mutual information of the following channel:*




*Here, the random variable $Z$ is independent of $X$ and $P[Z = 0] = P[Z = a] = \frac{1}{2}$ for some real number $a$. Further assume that $X$ is binary.*

**Exercise 6.5.** *Let $X$ and $Z$ be independent random variables taking values on $\{1, \ldots, n\}$ and $\{0, 1\}$, respectively, with $p_X(i) = q_i$ (for each $i$) and $p_Z(1) = p$. Define the random variable $Y := X \cdot Z$.*

a) *Write $H(Y)$ in terms of $H(X)$ and $H(Z)$.*

b) *Find $p$ and $\mathbf{q} = (q_1, \ldots, q_n)$ that maximize $H(Y)$.*

c) *Suppose $X$ and $Y$ are input and output of a channel. For a fixed $p \in [0, 1]$, what is the channel mutual information $I(p)$?*

**Exercise 6.6** (Composed channels)**.** *Consider two channels $W_1(y_1|x_1)$ and $W_2(y_2|x_2)$ with channel mutual information $I_1$ and $I_2$, respectively. Find the channel mutual information of the following composed channels.*

a) *Parallel composition:*

$$W(y_1, y_2|x_1, x_2) = W_1(y_1|x_1)W_2(y_2|x_2)\,.$$

b) *Selecting channel based on another input bit $z$:*

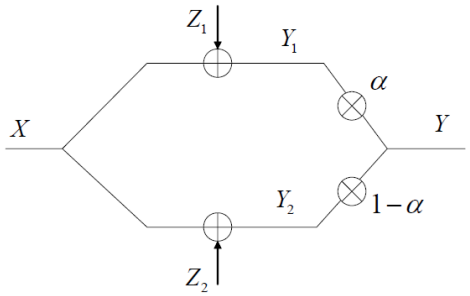
$$W(y|x, z) = \begin{cases} W_1(y|x) & \text{if } z = 0 \\ W_2(y|x) & \text{if } z = 1 \end{cases}.$$

*Here we assume that either $W_1(y|x) = 0$ or $W_2(y|x) = 0$ for all $x$ and $y$, that is, the channel outputs of the two channels are orthogonal.*

**Exercise 6.7.** *Consider a Gaussian channel shown below, in which the transmitted signal $X$ with $\mathbb{E}[X^2] \leq P$ is received by two antennas with $Y_1 = X + Z_1$ and $Y_2 = X + Z_2$ where $Z_1$ and $Z_2$ are independent with $\mathbb{E}[Z_i^2] = \sigma_i^2$ ($\sigma_1^2 < \sigma_2^2$). The signals at the two antennas are combined as*

$$Y = \alpha Y_1 + (1 - \alpha)Y_2$$

*before decoding, where $0 \leq \alpha \leq 1$.*



*Find the power-constrained channel mutual information of this channel as a function of* $\alpha \in [0,1]$.

# 7
# *Noisy channel coding*

**Intended learning outcomes:**

- You understand the formal setup of the noisy channel coding for discrete memoryless channels. You understand the conceptual difference between capacity and channel mutual information.
- You understand the concept of random codes and how to analyse their performance.
- You know asymptotic and one-shot bounds for noisy channel coding and can derive the former from the latter.
- You can determine if a source can be transmitted through a DMC using the source-channel separation theorem.

**Book reference:** Chapter 7 in Cover & Thomas[1].

[1] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991. ISBN 9780471748823. DOI: 10.1002/047174882X

## *7.1 Problem setup and definitions*

To quote Shannon from his pivotal paper[2]:

> The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.

[2] C. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 1948. DOI: 10.1002/j.1538-7305.1948.tb00917.x

The basic setup of the communication problem consists of a source that generates digital information which is to be reliably communicated to a destination through a channel, preferably in the most efficient manner possible. The destination could be spatially or temporally separated.

In this chapter we will first learn how to transmit a source that produces messages with uniform probability from some set of messages and then argue that the optimal strategy to transmit an arbitrary source is to first compress it (which makes it approximately uniform) and then send it over the channel. The latter is called the source-channel separation theorem since it allows to treat channel coding and source coding independently as two separate tasks, without loss of efficiency — at least in the asymptotic limit of large block lengths.

Let us now move on to a more operational description of the channel coding problem. As we have seen a noisy channel can be described by a conditional probability distribution $W_{Y|X}$. If such a channel can be used multiple times, without any memory effects, we speak of a *discrete memoryless channel* (DMC). We will not consider more complicated channels that change over time or have memory effects here, and thus our definition is restricted to the discrete memoryless case.

A *discrete memoryless channel* $\boldsymbol{W}$ from $\mathcal{X}$ to $\mathcal{Y}$ is characterised by a communication channel $W = W_{Y|X}$. For any $n \in \mathbb{N}$, it induces a channel $W^n$ that takes a sequence of input symbols $x^n \in \mathcal{X}^n$ to a sequence of output symbols $y^n \in \mathcal{Y}^n$ such that

$$P\big[Y^n = y^n \big| X^n = x^n\big] = W^n(y^n|x^n) = \prod_{i=1}^{n} W_{Y|X}(y_i|x_i)\,. \tag{7.1}$$

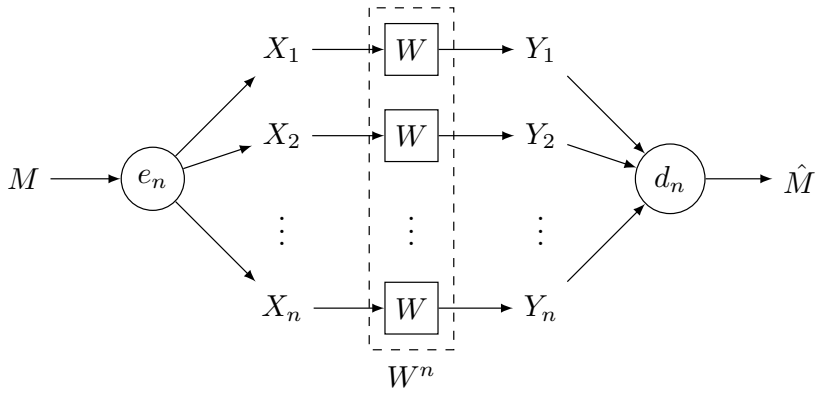



Figure 7.1: **The channel coding setup.** The figure depicts the setup for block length $n$. The encoder, $e_n$, takes a message $M$ and encodes it into $n$ channel input symbols, $X_1, X_2, \ldots, X_n$. The channels act independently on these input symbols. The channel outputs $Y_1, Y_2, \ldots, Y_n$ are decoded using the decoder, $d_n$, to an estimate $\hat{M}$ of $M$.

For our definition of codes, we can without loss of generality assume that the messages we are interested to send are in fact bit strings, i.e. $M \in \{0,1\}^L$. Any message set can obviously be encoded using bit strings, for example by using a compression algorithm!

A $(2^L, n)$-*channel code* for a DMC $\boldsymbol{W}$ is comprised of

- an encoder function $e_n : \{0,1\}^L \to \mathcal{X}^n$ and
- a decoder function $d_n : \mathcal{Y}^n \to \{0,1\}^L$.

Here $n$ is called the *block length* and $L$ is the message length in bits.

Consider the Markov chain $M \leftrightarrow X^n \leftrightarrow Y^n \leftrightarrow \hat{M}$ where the message $M$ follows a uniform distribution on $\{0,1\}^L$, the channel input is $X = e(M)$, the channel output $Y^n$ follows the distribution in (7.1),

and $\hat{M} = d(Y)$. The goal is to faithfully reproduce the message $M$ at the decoder, i.e., ideally we would like to have $\hat{M} = M$. However, for a probabilistic channel we cannot expect to always achieve this — instead, we want to control the probability of a decoding error.

An *average error* $(\epsilon, 2^L, n)$*-channel code* is an $(2^L, n)$-channel code such that the random variables defined above satisfy

$$P[M \neq \hat{M}] \leq \epsilon \,. \tag{7.2}$$

Here $\epsilon$ is the tolerated *average probability of error*.

We will come back to this in Section 7.2.4, where we will aim to bound the maximal (worst-case) probability of error instead.

The above formal definitions now allow us to define the concept of achievable rates and capacity of a DMC.

We say that a rate $R$ is a *achievable* on a DMC $\boldsymbol{W}$ if there exists a sequence $\{\epsilon_n : n \in \mathbb{N}\}$ and an average error $(\epsilon_n, 2^{\lceil nR \rceil}, n)$-channel code for all $n \in \mathbb{N}$ such that

$$\lim_{n \to \infty} \epsilon_n = 0 \tag{7.3}$$

Note that the size of the codebook increases exponentially with $n$, with the rate $R$ determining the exponent.

The channel capacity is the maximal rate that are achievable, i.e., the maximal rate for which such a sequence of codes with vanishing average probability of error exists.

The *channel capacity* of $\boldsymbol{W}$ is defined as

$$C(\boldsymbol{W}) := \sup \left\{ R \in \mathbb{R} : R \text{ is achievable on } \boldsymbol{W} \right\}. \tag{7.4}$$

The rate $R = 0$ is always achievable by a trivial code with just one message, so the channel capacity is certainly always non-negative. It can be equal to zero, however. One example for this is when the output is completely independent of the input, i.e., when $W(y|x) = Q(y)$ for all $x$ and some pmf $Q$ on the output alphabet. In this case we can clearly not even distinguish between two distinct messages at the receiver, and the average decoding error probability will be exactly $\frac{1}{2}$, independent of the encoder and decoder.

## 7.2 *Noisy channel coding theorem*

The main theorem of this chapter now relates the channel capacity of a DMC with the maximal mutual information of the underlying channel.

**Theorem 7.1** (Noisy channel coding theorem)**.** *For a DMC* $\boldsymbol{W}$ *with channel* $W$*, we have*

$$C(\boldsymbol{W}) = I(W)\,. \tag{7.5}$$

We will prove this theorem in several steps. First we will derive an upper bound on the cardinality of the message set that holds even for a single use of the channel, the so-called meta-converse. From this we will then prove the converse (upper bound on the rate) and finally show how this rate can be achieved using random codes.

### 7.2.1 *The meta-converse*

The task in noisy channel coding is to transmit a message reliably over a DMC. We continue to assume that the message is uniformly distributed over some set of messages, and thus treat the average error case. Same as with source coding, we will eventually consider this problem in an asymptotic scenario where the number of times the channel can be used, $n$, is taken to infinity. However, some results can conveniently be stated in a *one-shot setting*, without such a limit in mind, and we will do this here.

Let us consider the Markov chain given by $M \leftrightarrow X \leftrightarrow Y \leftrightarrow \hat{M}$ induced by a single use of the channel:

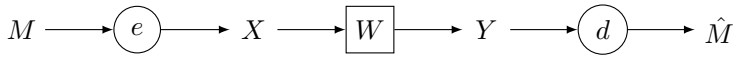



Figure 7.2: **One-shot setting for channel coding**: The channel is only used once, and we do not have any knowledge about its internal structure. Here, $M$ is distributed uniformly and the random variables $X$, $Y$ and $\hat{M}$ are induced by $e$, $W$ and $d$, respectively.

Note that in particular $W$ could be of the form $W^n$ as in Figure 7.1. In that case, Figure 7.2 in fact describes the same situation as Figure 7.1. Thus, the one-shot setting is more general than the asymptotic setting, even though on a first sight we seem to restrict ourselves to $n = 1$ uses of the channel.

Our first result is a bound on the cardinality of the message set. This result is called the *meta-converse* as it can be used to derive various different fundamental limits (or converse bounds). This result was only established (relatively) recently by Polyanskiy-Poor-Verdú[3]. So even though information theory (and in particular channel coding) is by now a very well-established discipline, some progress can still be made when it comes to simplifying mathematical proofs and presenting them in a unified way.

The idea is to relate the channel coding problem to binary hypoth-

[3] Yury Polyanskiy, H. Vincent Poor, and Sergio Verdú. Channel Coding Rate in the Finite Blocklength Regime. *IEEE Transactions on Information Theory*, 56(5):2307–2359, may 2010. DOI: 10.1109/TIT.2010.2043769

esis testing. More precisely, we consider the quantity (see Chapter 3)

$$\beta_1^*(\epsilon; P_{XY} \| P_X \times Q_Y) = \min_{A \subseteq \mathcal{X} \times \mathcal{Y}} \left\{ \sum_{(x,y) \notin A} P_X(x) Q_Y(y) : \sum_{(x,y) \in A} P_{XY}(x,y) \leq \epsilon \right\}, \tag{7.6}$$

where $P_{XY}$ is the joint distribution of channel inputs and outputs and $Q_Y$ is any distribution of $Y$. The idea is that we want to distinguish between a random variable $Y$ that has been produced by the channel and one that is completely independent of $X$. In other words, the null hypothesis is that $P_{XY}$ are in fact the channel input and output of our channel $W$, and the alternative hypothesis is that the output $Q_Y$ has been produced independently of the input $P_X$, i.e. that it is the output of a channel that is completely useless for information transmission.

We state our first one-shot result as follows.

**Proposition 7.2** (Meta-converse). *Any average error $(\epsilon, 2^L, 1)$-channel code for a channel $\boldsymbol{W}$ satisfies*

$$L \leq \max_{P_X} \min_{Q_Y} \log \frac{1}{\beta_1^*(\epsilon; P_{XY} \| P_X \times Q_Y)}, \tag{7.7}$$

*where $P_{XY}(x,y) = P_X(x) W_{Y|X}(y|x)$ is the joint distribution of channel input and $Q_Y$ is an arbitrary distribution on the output.*

We provide the proof here for the special case where the encoder and decoder are deterministic, and the encoder is injective. These assumptions make the proof a bit simpler but are not really restrictive. For any code using randomness shared between encoder and decoder there is a deterministic code performing at least equally well as is shown in Exercise 7.1. Injectivity of the encoder simply means that we do not assign the same codewords to two distinct messages, but clearly doing so would be sub-optimal as there would be no hope to distinguish these messages later.

One has to be a bitmore careful here. Do you see situations where this argument may fail?

*Proof of Proposition 7.2.* We first consider a fixed $(\epsilon, 2^L, 1)$-channel code that induces a distribution

$$P_X(x) = 2^{-L} \sum_{m \in \{0,1\}^L} \mathbf{1}\{e(m) = x\}. \tag{7.8}$$

on the channel input. By assumption, such a code must satisfy $P[M \neq \hat{M}] \leq \epsilon$. We now consider the hypothesis testing problem at hand, where $H_0$ is $P_{XY}$ and $H_1$ is $P_X \times Q_Y$ for some arbitrary distribution $Q_Y$. For this problem we take the test

$$\mathcal{A} = \{(x,y) \in \mathcal{X} \times \mathcal{Y} : x \neq e(d(y))\} \tag{7.9}$$

We can then compute the errors of the first and second kind for this test. This yields

$$\alpha(\mathcal{A}) = P_{XY}(\mathcal{A}) = P[X \neq e(d(Y))] = P[e(M) \neq e(\hat{M})] \tag{7.10}$$
$$\leq P[M \neq \hat{M}] \leq \epsilon\,. \tag{7.11}$$

Furthermore,

$$\beta(\mathcal{A}) = P_X \times Q_Y(\mathcal{A}^c) \tag{7.12}$$
$$= 2^{-L} \sum_{x,y} \sum_{m \in \{0,1\}^L} Q_Y(y) \mathbf{1}\{e(m) = x\}\, \mathbf{1}\{x = e(d(y))\} \tag{7.13}$$
$$= 2^{-L} \sum_{y} \sum_{m \in \{0,1\}^L} Q_Y(y)\, \mathbf{1}\{e(m) = e(d(y))\} \tag{7.14}$$
$$= 2^{-L} \sum_{y} Q_Y(y) = 2^{-L}\,, \tag{7.15}$$

where in the penultimate equality we used that there exists exactly one $m$ for which $e(m) = e(d(y))$ holds, which is ensured by the injectivity of the encoder.

Combining (7.11) and (7.15), we can deduce that $\beta_1^*(\epsilon; P_{XY} \| P_X \times Q_Y) \leq 2^{-L}$, and, optimising over $Q_Y$, we find

$$L \leq \min_{Q_Y} \log \frac{1}{\beta_1^*(\epsilon; P_{XY} \| P_X \times Q_Y)}\,. \tag{7.16}$$

Finally, since we do not know the distribution $P_X$ that the code induces — it depends on the specific encoding function $e$ used — we maximise over $P_X$ to get a bound that holds for all $(\epsilon, 2^L, 1)$-channel codes. □

### 7.2.2 *Proof of strong converse*

In the following we will show that $C(\boldsymbol{W}) \leq I(W)$. In fact, we will show something much stronger. We will show that for any sequence of average error $(\epsilon, 2^{\lceil Rn \rceil}, n)$-channel codes for a DMC $\boldsymbol{W}$ with $\epsilon \in [0, 1)$ fixed, we must have $R \leq I(W)$.[4] Hence, even if we allow for any non-zero success probability asymptotically, the maximal rate is still bounded by the channel mutual information. This is what is called a strong converse for channel coding. The proof strategy we follow here is inspired by some of our own work[5], and if taken to its conclusion can yield tight higher-order expansions of the channel capacity. However, here we are only interested in the capacity, which allows us to simplify the argument quite a bit.

[4] Obviously, if we set $\epsilon = 1$ then anything goes and there cannot be any bound on the maximal rate.

[5] Marco Tomamichel and Vincent Y. F. Tan. A Tight Upper Bound for the Third-Order Asymptotics for Most Discrete Memoryless Channels. *IEEE Transactions on Information Theory*, 59(11):7041–7051, nov 2013. DOI: 10.1109/TIT.2013.2276077

**Proposition 7.3** (Strong converse for channel coding)**.** *Let* $\boldsymbol{W}$ *be a*

*DMC with channel $W$. Then, for any sequence of $(\epsilon_n, 2^{\lceil nR \rceil}, n)$-codes with $\limsup_{n\to\infty} \epsilon_n < 1$, we must have $R \leq I(W)$.*

This implies that even if we allow for an error $\epsilon < 1$, we still cannot achieve any rate exceeding the channel mutual information. It also directly implies the converse part of the noisy channel coding theorem, Theorem 7.1, since it ensures that $C(\mathbb{W}) \leq I(W)$.

Before we prove this statement we first derive a relaxation of the meta-converse in terms of the information spectrum relative entropy. For this we will need the following lemma:

**Lemma 7.4.** *For any pmfs $P_X$ and $Q_Y$ and channel $W_{Y|X}$ it holds that*

$$D_s^\epsilon\big(P_{XY}\big\|P_X \times Q_Y\big) \leq \max_{x\in\mathcal{X}} D_s^\epsilon\big(W(\cdot|x)\big\|Q_Y(\cdot)\big)\,. \tag{7.17}$$

*Proof.* We first introduce the log-likelihood ratio conditioned on $x$, the random variable

$$Z_x = \log \frac{W_{Y|X}(Y|x)}{Q_Y(Y)}, \tag{7.18}$$

where $Y$ is distributed according to the law $W_{Y|X=x}$. We then write

$$D_s^\epsilon(W(\cdot|x)\big\|Q_Y(\cdot)) = \sup\left\{R \in \mathbb{R} : P[Z_x \leq R] \leq \epsilon\right\}. \tag{7.19}$$

Moreover, using the law of total probability, we find

$$\begin{aligned} &P\left[\log \frac{P_{XY}(X,Y)}{P_X(X)Q_Y(Y)} \leq R\right] \\ &\quad= \sum_x P_X(x) W_{Y|X=x}\left[\log \frac{W_{Y|X}(Y|x)}{Q_Y(Y)} \leq R\right] &(7.20)\\ &\quad= \sum_x P_X(x) P\left[Z_x \leq R\right]. &(7.21)\end{aligned}$$

Plugging this into the definition of the information spectrum relative entropy, we find

$$\begin{aligned} D_s^\epsilon\big(P_{XY}\big\|P_X \times Q_Y\big) &= \sup\left\{R \in \mathbb{R} : \sum_x P_X(x) P\left[Z_x \leq R\right] \leq \epsilon\right\} &(7.22)\\ &\leq \sup\left\{R \in \mathbb{R} : P\left[Z_{x^*} \leq R\right] \leq \epsilon\right\} &(7.23)\\ &\leq \max_{x\in\mathcal{X}} \sup\left\{R \in \mathbb{R} : P\left[Z_x \leq R\right] \leq \epsilon\right\}. &(7.24)\end{aligned}$$

To establish the first inequality we used that there must (at least) exist one $x^*$ for which the probability $P[Z_{x^*} \leq R]$ does not exceed its expectation over $X$. Hence, by relaxing the condition in the supremum we get an upper bound on the quantity. The final inequality then simply follows by maximising over all $x$ instead, removing the dependence on $X^*$, which we don not know. □

*Proof of Proposition 7.3.* Consider a sequence of $(\epsilon_n, 2^{\lceil Rn \rceil}, n)$-codes as in the statement of the result. Since we have that $\epsilon_\infty := \limsup_{n\to\infty} \epsilon_n < 1$ we may choose $\epsilon \in (\epsilon_\infty, 1)$ and by definition of the limit, there exists an $N \in \mathbb{N}$ such that for all $n \geq N$ there exists an $(\epsilon, 2^{\lceil nR \rceil}, n)$-code in the sequence. Recall that

$$P_{X^nY^n}(x^n, y^n) = P_{X^n}(x^n) \prod_{i=1}^{n} W_{Y|X}(y_i|x_i) \tag{7.25}$$

is the joint distribution of channel inputs and outputs when the super-channel $W^n$ is applied to an arbitrary input distributions $P_{X^n}$. By the meta-converse for the super-channel $W^n$, we have

$$\begin{aligned} nR &\leq \lceil nR \rceil && (7.26)\\ &\leq \max_{P_{X^n}} \min_{Q_{Y^n}} \log \frac{1}{\beta_1^*(\epsilon; P_{X^nY^n} \| P_{X^n} \times Q_{Y^n})}, && (7.27) \end{aligned}$$

where the minimisation is now over joint distributions $Q_{Y^n}$ of all channel outputs. However, since we are interested in an upper bound we can choose this distribution to be i.i.d. of the form $Q_Y^n$, where $Q_Y$ is the output distribution that minimises the expression in (6.21), i.e., it satisfies

$$D(W(\cdot|x)\|Q_Y) \leq I(W) \quad \forall x \in \mathcal{X}. \tag{7.28}$$

Hence, using Lemma 3.9 and Lemma 7.4, we find

$$\begin{aligned} nR &\leq \max_{P_{X^n}} \log \frac{1}{\beta_1^*(\epsilon; P_{X^nY^n} \| P_{X^n} \times Q_Y^n)} && (7.29)\\ &\leq \max_{P_{X^n}} D_s^{\epsilon+\delta}(P_{X^nY^n} \| P_{X^n} \times Q_Y^n) + \log \frac{1}{\delta} && (7.30)\\ &\leq \max_{x^n \in \mathcal{X}^n} D_s^{\epsilon+\delta}\big(W^n(\cdot|x^n) \big\| Q_Y^n(\cdot)\big) + \log \frac{1}{\delta}, && (7.31) \end{aligned}$$

where we have chosen $\delta$ such that $\epsilon + \delta < 1$. In the last step we simply noted that the expression no longer depends on the input distribution $P_{X^n}$. It thus remains to analyse the information spectrum relative entropy in (7.31) as a function of $x^n \in \mathcal{X}^n$. In particular, the probability

$$P\left[\sum_{i=1}^{n} Z_{x_i}^i \leq R\right], \quad \text{where } Z_x^i \text{ are independent rvs,} \tag{7.32}$$

given by $Z_x^i = \log \frac{W_{Y|X}(Y|x)}{Q_Y(Y)}$ with $Y$ distributed according to $W_{Y|X=x}$. Hence, the $Z_{x_i}^i$ are independent but not identically distributed. Using this, we can write

$$D_s^{\epsilon+\delta}\big(W^n(\cdot|x^n)\big\|Q_Y^n\big) = \sup\left\{R \in \mathbb{R} : P\left[\sum_{i=1}^{n} Z_{x_i}^i \leq R\right] \leq \epsilon + \delta\right\}. \tag{7.33}$$

The first thing we note is that this expression actually does not depend on all the properties of $x^n$, but only the frequency in which the letters of $\mathcal{X}$ occur in $x^n$, which makes it suitable to an application of the Method of Types. However, we will not need this here.

The random variable in (7.32) is a sum of independent (but not identically distributed) random variables. Let us compute its expectation and variance, which are

$$\mathbb{E}\left[\sum_{i=1}^{n} Z_{x_i}^i\right] = \underbrace{\sum_{i=1}^{n} D\left(W_{Y|X}(\cdot|x_i)\|Q_Y\right)}_{=:J} \tag{7.34}$$

$$\operatorname{Var}\left[\sum_{i=1}^{n} Z_{x_i}^i\right] = \sum_{i=1}^{n} \operatorname{Var}\left[\log \frac{W_{Y|X}(Y_i|x_i)}{Q_Y(Y_i)}\right] \tag{7.35}$$

$$\leq \underbrace{n \max_{x\in\mathcal{X}} \operatorname{Var}\left[\log \frac{W_{Y|X}(Y_i|x)}{Q_Y(Y_i)}\right]}_{=:\,\sigma^2}, \tag{7.36}$$

where $\sigma^2$ is a constant (depending on the channel, but not the input sequence $x^n$). So, for any $\nu > 0$, Chebyshev's inequality yields

$$P\left[\sum_{i=1}^{n} Z_{x_i}^i \geq J + \sqrt{n\nu}\right] \leq \frac{n\sigma^2}{n\nu} = \frac{\sigma^2}{\nu}, \tag{7.37}$$

or, equivalently,

$$P\left[\sum_{i=1}^{n} Z_{x_i}^i \leq J + \sqrt{n\nu}\right] \geq 1 - \frac{\sigma^2}{\nu}, \tag{7.38}$$

If we just choose $\nu$ large enough, we can make the left-hand side larger than $\epsilon + \delta$. Hence, we can deduce that, for such a $\nu$,

$$D_s^{\epsilon+\delta}\left(W^n(\cdot|x^n)\middle\|Q_Y^n(\cdot)\right) \leq J + \sqrt{n\nu} \leq nI(W) + \sqrt{n\nu}\,, \tag{7.39}$$

where the second inequality is due to (7.28). This upper bound now no longer depends on $x^n$. Plugging it into Eq. (7.31) and dividing both sides by $n$, we find

$$R \leq I(W) + \frac{\nu}{\sqrt{n}} + \frac{1}{n}\log\frac{1}{\delta} \tag{7.40}$$

Taking the limit $n \to \infty$, we can conclude that the inequality $R \leq I(W)$ must hold for any such sequence of codes. □

### 7.2.3 *Proof of achievability and random codes*

We will need the following technical lemma. (Our proof is inspired by the analysis of Hayashi and Nagaoka in the domain of quantum information theory[6].)

[6] Masahito Hayashi and Hiroshi Nagaoka. General Formulas for Capacity of Classical-Quantum Channels. *IEEE Transactions on Information Theory*, 49(7):1753–1768, jul 2003. DOI: 10.1109/TIT.2003.813556

**Lemma 7.5.** *Let $t \geq 0$ and $s \in [0,1]$. Then, $1 - \frac{s}{s+t} \leq 1 - s + t$.*

*Proof.* We may rewrite the statement as

$$0 \leq t - s + \frac{s}{s+t}\,. \tag{7.41}$$

If $t \geq s$ this is trivially true, so in the following we may assume $s > 0$. If $t < s$ we use the convexity of the function $f(t) = \frac{s}{s+t}$ to bound it with its tangent at $t = 0$. This yields

$$\frac{s}{s+t} \geq 1 - \frac{t}{s} = \frac{s-t}{s} \geq s - t, \tag{7.42}$$

where the second inequality follows since $s \in (0,1]$ and $s - t \geq 0$. □

We again first analyse the channel coding problem in the one-shot setting where the channel is only used once.

**Proposition 7.6.** *For any $\epsilon, \delta \in (0,1)$ such that $\epsilon + \delta < 1$ there exists an $(\epsilon + \delta, 2^L, 1)$-channel code for a stochastic map $W_{Y|X}$ as long as the code parameters satisfy*

$$L \leq \log \frac{\delta}{\beta_1^*(\epsilon; P_{XY} \| P_X \times P_Y)} \tag{7.43}$$

*for some pmf $P_X$ on $\mathcal{X}$.*

*Proof.* We now construct a random code for a single use of the channel. First, we fix any distribution $P_X$. From this we generate $|M|$ codewords independently by picking them from the distribution $P_X$, i.e. the output of the encoder, $E(m)$, is itself a random variable following the distributions $P_X$ for each message $m$. The decoder is constructed as follows. Consider the binary hypothesis testing problem between $H_0 : P_{XY}$ and $H_1 : P_X \times P_Y$. By definition of $\beta_1^*(\epsilon; P_{XY} \| P_X \times P_Y)$, there exists a subset $\mathcal{A} \subset \mathcal{X} \times \mathcal{Y}$ that satisfies

$$P_{XY}(\mathcal{A}^c) \leq \epsilon \quad \text{and} \quad (P_X \times P_Y)(\mathcal{A}) = \beta_1^*(\epsilon; P_{XY} \| P_X \times P_Y)\,. \tag{7.44}$$

From this we construct the sets $\mathcal{A}_x = \{y \in \mathcal{Y} : (x,y) \in \mathcal{A}\}$ for all $x \in \mathcal{X}$. For a fixed encoder $E = e$, the decoder is also probabilistic. Given a channel output $y$ it assigns $\hat{M} = m$ with probability

$$P[\hat{M} = m | Y = y, E = e] = \frac{\mathbf{1}\{y \in \mathcal{A}_{e(m)}\}}{\sum_{m'} \mathbf{1}\{y \in \mathcal{A}_{e(m')}\}} \tag{7.45}$$

$$= \frac{\mathbf{1}\{y \in \mathcal{A}_{e(m)}\}}{\mathbf{1}\{y \in \mathcal{A}_{e(m)}\} + \sum_{m' \neq m} \mathbf{1}\{y \in \mathcal{A}_{e(m')}\}}\,. \tag{7.46}$$

Let us now analyse the probability of error for this code, first for a fixed set of codewords (or fixed encoder $e$) and fixed message $m$.

$$\begin{aligned} &P[M \neq \hat{M} | M = m, E = e] \\ &\quad = 1 - \sum_y W_{Y|X}(y|e(m)) P[\hat{M} = m | Y = y, E = e] && (7.47) \\ &\quad = \sum_y W_{Y|X}(y|e(m)) \left( 1 - \frac{\mathbf{1}\{y \in \mathcal{A}_{e(m)}\}}{\mathbf{1}\{y \in \mathcal{A}_{e(m)}\} + \sum_{m' \neq m} \mathbf{1}\{y \in \mathcal{A}_{e(m')}\}} \right) && (7.48) \end{aligned}$$

We can now use Lemma 7.5 to bound this as

$$\begin{aligned} &P[M \neq \hat{M} | M = m, E = e] && (7.49) \\ &\quad \leq \sum_y W_{Y|X}(y|e(m)) \left( \mathbf{1}\{y \notin \mathcal{A}_{e(m)}\} + \sum_{m' \neq m} \mathbf{1}\{y \in \mathcal{A}_{e(m')}\} \right) . && (7.50) \end{aligned}$$

We may now take the average over all encoders $e$, so that $E(m)$ and $E(m')$ for $m \neq m'$ are independent and both follow the distribution $P_X$. This gives the following bound

$$\begin{aligned} &P[M \neq \hat{M} | M = m] && (7.51) \\ &\leq \sum_{x,x'} P_X(x) P_X(x') \sum_y W(y|x) \left( \mathbf{1}\{y \notin \mathcal{A}_x\} + \sum_{m' \neq m} \mathbf{1}\{y \in \mathcal{A}_{x'}\} \right) && (7.52) \\ &= \sum_{x,y} P_X(x) W_{Y|X}(y|x) \left( \mathbf{1}\{y \notin \mathcal{A}_x\} + \underbrace{(2^L - 1)}_{\leq 2^L} \sum_{x' \in \mathcal{X}} P_X(x') \mathbf{1}\{y \in \mathcal{A}_{x'}\} \right) && (7.53) \end{aligned}$$

and we note that the bound no longer depends on the choice of $m$, i.e. Eq. (7.53) is in fact an upper bound on $P[M \neq \hat{M}]$.

Let us now investigate the two summands in (7.53) individually. We first observe that

$$\begin{aligned} \sum_{x,y} P_X(x) W_{Y|X}(y|x) \mathbf{1}\{y \notin \mathcal{A}_x\} &= P_{XY}[(x,y) \notin \mathcal{A}] && (7.54) \\ &= P_{XY}[A^c] \leq \epsilon && (7.55) \end{aligned}$$

by definition of the sets $\mathcal{A}_x$ and $\mathcal{A}$. We can also evaluate

$$\begin{aligned} &\sum_{x,y} P_X(x) W_{Y|X}(y|x) \sum_{x'} P_X(x') \mathbf{1}\{y \in \mathcal{A}_{x'}\} \\ &\quad = \sum_y P_Y(x) \sum_{x'} P_X(x') \mathbf{1}\{(x', y) \in \mathcal{A}\} && (7.56) \\ &\quad = (P_X \times P_Y)[\mathcal{A}] && (7.57) \\ &\quad = \beta_1^*(\epsilon; P_{XY} \| P_X \times P_Y) . && (7.58) \end{aligned}$$

Summarising this, we find that

$$P[M \neq \hat{M}] \leq \epsilon + 2^L \beta_1^*(\epsilon; P_{XY} \| P_X \times P_Y)\,. \tag{7.59}$$

So, in particular, as long as we choose $2^L \leq \delta \cdot \beta_1^*(\epsilon; P_{XY} \| P_X \times P_Y)^{-1}$, we achieve $P[M \neq \hat{M}] \leq \epsilon + \delta$, as required.

Finally, since this bound holds on average over all choices of encoders $e$, there exists (at least) one encoder that satisfies $P[M \neq \hat{M} | E = e] \leq \epsilon + \delta$ as well. This is the code we are looking for. □

With this result at hand, we can complete the proof of Theorem 7.1, showing that any rate $R < I(W)$ is achievable.

*Achievability of Theorem 7.1.* We consider the one-shot results applied to the super-channel $W^n$. Proposition 7.6 stipulates that there exists a code with $2^{\lceil nR \rceil}$ codewords and error $2\epsilon$ as long as

$$\lceil nR \rceil \leq \log \frac{\epsilon}{\beta_1^*(\epsilon_n; P_{X^n Y^n} \| P_{X^n} \times P_{Y^n})} \tag{7.60}$$

for some input distribution $P_{X^n} \in \mathcal{P}(\mathcal{X}^n)$. Hence, removing the ceiling function and dividing by $n$ on both sides, as long as

$$R \leq \frac{1}{n} \left( \log \frac{1}{\beta_n^*(\epsilon; P_{XY} \| P_X \times P_Y)} + \log \epsilon - 1 \right). \tag{7.61}$$

We further choose $P_{X^n}$ to be i.i.d. and $P_X$ the maximiser in the definition of the channel mutual information, i.e. $I(X:Y)_P = I(W)$ to make our analysis simpler. We can then use the Chernoff-Stein Lemma (cf. Theorem 3.8) to show that

$$\lim_{n\to\infty} \frac{1}{n} \log \frac{1}{\beta_\epsilon^*(P_{XY}^n \| P_X^n \times P_Y^n)} = \lim_{n\to\infty} \frac{1}{n} \log \frac{1}{\beta_n^*(\epsilon; P_{XY} \| P_X \times P_Y)} \tag{7.62}$$

$$= D(P_{XY} \| P_X \times P_Y) \tag{7.63}$$

$$= I(X:Y)_P = C(W). \tag{7.64}$$

And thus, we can conclude that if $R < I(W)$ is strictly smaller than the channel mutual information, then for any error $\epsilon > 0$ the condition (7.61) will be satisfied for sufficiently large $n$. Hence, there exists a sequence of codes at rate $R$ with average probability of error $2\epsilon$. Since this is true for any $\epsilon > 0$, we can indeed construct a sequence of codes with vanishing probability of error at this rate, and it is thus achievable. Taking the supremum over all such rates yields our achievability bound $C(\boldsymbol{W}) \geq I(W)$. □

### 7.2.4 *Maximum probability of error*

Consider a code with $|M|$ codewords. So far we have used the *average probability of error* as a metric for our codes, namely we required that

$$P[\hat{M} \neq M] = \sum_{m \in \{0,1\}^L} 2^{-L}\, P[\hat{M} \neq m | M = m] \tag{7.65}$$

vanishes asymptotically. Sometimes we would however like to impose an even stricter condition, namely that the *maximum probability of error*, given by

$$\max_{m\in\{0,1\}^L} P[\hat{M} \neq m | M = m], \tag{7.66}$$

vanishes asymptotically. Because the condition is stricter our converse bounds still hold even with this new definition of error; however, the random codes we constructed so far do not necessarily lead to a small maximum probability of error.

The following lemma allows us to construct codes that overcome this. It shows that, accepting a loss of just one bit of message, we can promote any $\epsilon$ average error code to a $2\epsilon$ maximum error code.

**Proposition 7.7.** *Given an average error $(\epsilon, 2^L, 1)$-channel code, we can construct a maximum error $(2\epsilon, 2^{L-1}, 1)$-channel code.*

*Proof.* The construction uses expurgation of bad codewords. By definition of the $(\epsilon, 2^L, 1)$-average error channel code, we have

$$\sum_{m\in\{0,1\}^L} 2^{-L} P[\hat{M} \neq m | M = m] \leq \epsilon \tag{7.67}$$

Hence, there must be a subset $M_g \subseteq \{0,1\}^L$ of size at least $2^{L-1}$ with

$$P[\hat{M} \neq m | M = m] \leq 2\epsilon \qquad \forall\, m \in M_g \tag{7.68}$$

as otherwise the inequality in Eq. (7.67) cannot hold. The codewords in $M_g$ constitute an $(2\epsilon, \frac{|M|}{2}, 1)$-maximum error channel code. □

We now note that this expurgation is not affecting the codes asymptotically. Let us say a rate $R$ is achievable with average error. Then, for any rate $R' < R$, for large enough $n$ we have $2^{\lceil nR \rceil - 1} \geq 2^{\lceil nR' \rceil}$. Hence, the rate $R'$ is achievable with maximum error. Since the capacity is defined as the supremum over all achievable rates, the capacity remains the same.

## 7.3 Source-channel separation theorem

We have until now covered the case where a message that is uniformly chosen from a set needs to be transmitted through the noisy channel. Does anything change when instead we want to transmit a general source? The setting is the same as with channel coding, except that now for each block length $n$ we want to transmit a memoryless source given by i.i.d. $Z^n = (Z_1, Z_2, \ldots, Z_n)$. A code for block length $n$ is given by an encoder $e_n : \mathcal{Z}^n \to \mathcal{X}^n$ and a decoder

$d_n : \mathcal{Y}^n \to \mathcal{Z}^n$ and our goal is to find a sequence of such codes that satisfy

$$\lim_{n\to\infty} P[\hat{Z}^n \neq Z^n] \to 0\,. \tag{7.69}$$

Here we want to show the following theorem:

**Theorem 7.8.** *Given a DMS* $\mathbf{Z}$ *and DMC* $\mathbf{W}$*, there exists a sequence of codes satisfying* (7.69) *if* $H(Z) < I(W)$*. Moreover, if* $H(Z) > I(W)$ *such a sequence of codes cannot exist.*

When $H(Z) < I(W)$ we can simply compress the source at a rate $R = H(Z) + \mu$ and then transmit it over the channel at the same rate $R = I(W) - \mu$, where we choose $\mu = \frac{1}{2}(I(W) - H(Z))$. That is, we first apply the encoder for source compression, transmit the compressed source through the channel using a maximum probability of error channel code, and finally decompress the source at the receiver. The error of such a scheme is simply the sum of the individual errors of the source compression code and the channel code, both of which vanish asymptotically as shown in the source and channel coding theorems, respectively.

The second statement of this theorem, which is conceptually more interesting, shows that such a separate treatment of compression and channel coding is in fact optimal (at least when we only look at the first order asymptotics). We will only give a formal proof of the second statement. We will need the following lemma for this purpose, which ensures the additivity of the channel mutual information.

**Lemma 7.9.** *Let* $W_1$ *and* $W_2$ *be two channels. Then,*

$$I(W_1 \times W_2) = I(W_1) + I(W_2)\,. \tag{7.70}$$

*Proof.* We first note that for two channel inputs $X_1$ and $X_2$ following any distribution $P_{X_1X_2}$ and two channel outputs $Y_1$ and $Y_2$ produced by two channels $W_1$ and $W_2$ applied to $X_1$ and $X_2$, respectively, we

have

$$
\begin{aligned}
I(X_1X_2 : Y_1Y_2) &= \\
&= H(Y_1Y_2) - H(Y_1Y_2|X_1X_2) && (7.71)\\
&\leq H(Y_1) + H(Y_2) \\
&\quad - \sum_{x_1,x_2\in\mathcal{X}} P_{X_1X_2}(x_1,x_2)\, \underbrace{H(Y_1Y_2|X_1=x_1, X_2=x_2)}_{=H(Y_1|X_1=x_1)+H(Y_2|X_2=x_2)} && (7.72)\\
&= H(Y_1) + H(Y_2) - H(Y_1|X_1) - H(Y_2|X_2) && (7.73)\\
&= I(X_1 : Y_1) + I(X_2 : Y_2) && (7.74)\\
&\leq I(W_1) + I(W_2)\,. && (7.75)
\end{aligned}
$$

Here we used sub-additivity of entropy and the fact that $Y_1$ and $Y_2$ are independent once we condition on $X_1$ and $X_2$. We conclude that this inequality holds in particular also for the distribution $P_{X_1X_2}$ achieving $I(W_1 \times W_2)$. The other direction and thus equality clearly holds since we can always just restrict ourselves to product distributions when optimising $I(W_1 \times W_2)$, and, thus

$$
\begin{aligned}
I(W_1 \times W_2) &\geq \max_{P_{X_1},P_{X_2}} I(X_1X_2 : Y_1Y_2) && (7.76)\\
&= \max_{P_{X_1},P_{X_2}} I(X_1 : Y_1) + I(X_2 : Y_2) = I(W_1) + I(W_2)\,. && (7.77)
\end{aligned}
$$

□

*Proof of Theorem 7.8.* Assume $H(Z) - I(W) = \nu > 0$. If there is a sequence of codes with asymptotically vanishing error then for every $\epsilon > 0$ there must be a block length $n$ such that $P[\hat{Z}^n \neq Z^n] \leq \epsilon$. For such a code, by Fano's inequality, we have

$$H(Z^n) - I(Z^n : Y^n) = H(Z^n|Y^n) \leq H(Z^n|\hat{Z}^n) \leq 1 + \epsilon\, n \log|Z| \quad (7.78)$$

We can now evaluate $H(Z^n) = nH(Z)$ since the source is i.d.d., and furthermore

$$I(Z^n : Y^n) \leq I(X^n : Y^n) \leq I(W^n) = nI(W)\,. \quad (7.79)$$

To verify the first inequality we simply note that $Z^n \to X^n \to Y^n$ form a Markov chain, which implies that $I(Z^n : Y^n) \leq I(X^n : Y^n)$. Maximising $I(X^n : Y^n)$ over all input distributions then yields the inequality. The last inequality follows from Lemma 7.9, which can be used to verify that $I(W^n) = nI(W)$ by induction.

Finally, combining Eqs. (7.78) and (7.79) yields

$$\epsilon \log|Z| \geq H(Z) - I(W) - \frac{1}{n} = \nu - \frac{1}{n} \quad (7.80)$$

but since for large enough $n$ the term on the right-hand side is strictly positive, $\epsilon$ is bounded away from zero, leading to a contradiction. □

## 7.4 *Channel coding with power constraints*

In this chapter we have only dealt with discrete channels — but what happens in the continuous case and in particular for the AWGN channel? There, it remains to show that the channel capacity equals the power-restricted channel mutual information, i.e.,

$$C(\boldsymbol{W}) = \max_{p_X : \mathbb{E}[X^2] \leq P} I(X : Y) \tag{7.81}$$

For the converse, we can actually largely build on the proof we already have — we will thus only sketch the argument here. We will ignore all technical aspects that come from the fact that we are now dealing with pdfs instead of pmfs and focus on the main idea. In the meta-converse, in the last step we introduced a maximisation over all channel input distributions: if we have restrictions on which codewords are allowed, we can also restrict the distribution there. Thus, when we apply the meta-converse for $n$ channels, we now get

$$R \leq \min_{q_{Y^n}} \max_{p_{X^n}} \frac{1}{n} \log \frac{1}{\beta_1^*(\epsilon; p_{X^n Y^n} \| p_{X^n} \times q_{Y^n})}\,, \tag{7.82}$$

where we optimise over pdfs $p_{X^n}$ with support only on codewords $x^n$ that satisfy the constraint $\frac{1}{n}\sum_{i=1}^n x_i^2 \leq P$. This will help us in Eq. (7.17), where we can now restrict the optimisation over sequences $x^n$ with the above property as well. It remains now only to note that the empirical distributions corresponding to these sequences satisfy

$$\mathbb{E}[X^2] = \sum_x P_X^{x^k}(x) x^2 = \frac{1}{n}\sum_{i=1}^n x_i^2 \leq P\,. \tag{7.83}$$

The converse can thus be generalised along these lines to the continuous case, although in our treatment here we neglected various technicalities that come from dealing with pdfs instead of pmfs.

A very rough sketch can also be drawn up for achievability. The critical part here is that we choose codewords $X^n$ at random using the i.i.d. law $p_X^n$ and for some distribution with $\mathbb{E}[X^2] = P - \eta$ and $\eta > 0$, their power consumptions satisfies

$$\frac{1}{n}\sum_{i=1}^n X_i^2 \to \mathbb{E}[X^2] = P - \eta, \tag{7.84}$$

as $n \to \infty$ and by the weak law of large numbers we can thus argue that

$$P\left[\frac{1}{n}\sum_{i=1}^n X_i^2 \leq P\right] \to 1\,. \tag{7.85}$$

Thus, with high probability the codewords constructed in our random coding scheme satisfy the power constraint. In case an invalid

codeword is chosen by the random process we simply discard it and count this as an error. Since this argument works for any $\eta > 0$, we can eventually maximise over input distributions that satisfy the power constraint $\mathbb{E}[X^2] = P$.

## 7.5 Exercises

**Exercise 7.1.** *Suppose that in the definition of the $(2^{nR}, n)$ code for the DMC $p(y|x)$, we allow the encoder and the decoder to use random mappings. Specifically, let $K$ be an arbitrary random variable shared by sender and receiver and independent of the message $M$ and the channel. The encoder generates a codeword $x^n(m,k)$, and the decoder generates an estimate $\hat{m}(y^n,k)$ depending on the value of $k$. Show that this randomization does not increase the capacity of the DMC.*

**Hint:** Write $H(M) = I(M : Y^n, W) + H(M|Y^n, W)$, and use Fano's inequality.

**Exercise 7.2.** *Consider that there is a binary symmetric channel (BSC) with crossover probability $\epsilon$. We use a coding scheme on this channel that encodes messages $a_1$ and $a_2$ as* 000 *and* 111*, respectively. On the other hand, we use a decoding scheme that relies on the majority principle, i.e.,* $001 \mapsto 0$ *and* $011 \mapsto 1$.

a) *Compose the above encoder, the BSC, and the above decoder into a new channel. Describe the input/output alphabets and the channel rule (the conditional probability of outputs given inputs) of this new channel.*

b) *Assume the original BSC has crossover probability $\epsilon = 0.1$. What is its channel mutual information? Compare it to the channel mutual information of the composed channel.*

c) *Suppose that we have 4 input messages (each to be sent with equal probability). Find a code of length 3 that minimises the average decoding error, and compute it.*

d) *Consider a binary erasure channel (BEC) with erasure probability $\epsilon = 0.1$. Suppose we encode the messages $a_1$ and $a_2$ as* 000 *and* 111*, respectively. On the decoder side, if we receive $\perp\perp\perp$, we will randomly assign it to $a_1$ or $a_2$ with equal probability; otherwise, we decode according to the survived symbol, i.e.,* $0\perp\perp \mapsto a_1$ *and* $\perp\perp 1 \mapsto a_2$*. What is the average decoding error in this case?*

**Exercise 7.3** (Feinstein's bound)**.** *Let $W_{Y|X}$ be a channel from input set $\mathcal{X}$ to output set $\mathcal{Y}$.*

a) *Suppose we use the channel* once*. Show that there exist a code with $M$ codewords with average probability of error $\epsilon$ satisfying*

$$\epsilon \leq P\left[\log \frac{W_{Y|X}(Y|X)}{P_Y(Y)} \leq \log M + \gamma\right] + 2^{-\gamma}.$$

*for any choice of $\gamma > 0$ and any input distribution $P_X$ where $P_Y(y) = \sum_x W_{Y|X}(y|x)P_X(x)$. A stronger version of this bound (for maximum error) was shown by Feinstein.*

b) *Use a) to prove achievability of the channel coding theorem for DMCs.*

c) *Again consider the setup in a). Let*

$$V := \mathrm{Var}\left[\log \frac{P^*_{Y|X}(Y|X)}{P^*_Y(Y)}\right]$$

*be evaluated at a capacity-achieving input distribution $P^*_X$. Show that, by the central limit theorem, there exists a sequence of codes indexed by blocklength $n$, with sizes $M_n$ satisfying*

$$\log M_n = nC + \sqrt{nV}\Phi^{-1}(\epsilon) + o(\sqrt{n})$$

*such that the average error probability is no larger than $\epsilon + o(1)$.*

**Hint:** Generate codewords independently according to $P_X$. Instead of using typical set for decoding, use $\hat{m} \in \{1, \ldots, M\}$ as the transmitted message if it is the unique one satisfying

$$\log \frac{W_{Y|X}(y|x(\hat{m}))}{P_Y(y)} \geq \log M + \gamma.$$

If there is no unique $\hat{m}$ satisfying the above condition, declare an error.

**Exercise 7.4.** *We consider the channel coding problem for a discrete channel with memory. The channel state $s \in \mathcal{S}$ is i.i.d. according to some pmf $P_S$ and known to the decoder but not the encoder. The behaviour of the channel is then determined by a conditional pmf $W_{Y|XS}$. We are first interested in a one-shot converse bound. Consider an arbitrary code for sending $M$ distinct messages over $W$. Such a code is given by an encoder $e : [M] \to \mathcal{X}$ and a decoder $d : \mathcal{Y} \times \mathcal{S} \to [M]$. As usual we consider the case where $M$ follows a uniform distribution and let $\varepsilon$ denote the average probability of error.*

a) *Derive the following bound, a meta-converse for this problem:*

$$|M| \leq \max_{P_X} \min_{Q_{Y|S}} \frac{1}{\beta^*_\varepsilon(P_{XYS} \| P_X \times Q_{YS})}\,.$$

*where $P_{XYS}(x,y,s) = P_X(x)P_S(s)W_{Y|XS}(y|x,s)$ and $Q_{YS}(y,s) = Q_{Y|S}(y|s)P_S(s)$.*

b) *Further relax this bound and show that, for any $\delta \in (0, 1-\varepsilon)$,*

$$\log|M| \leq \min_{Q_{Y|S}} \max_{x \in \mathcal{X}} D_s^{\epsilon+\delta}(P_{YS|X=x} \| Q_{YS}) + \log\frac{1}{\delta}\,.$$

c) *Consider the channel mutual information $I(W) := \max_{P_X} I(X : Y|S)$ with $P_{XYS}$ as defined above. Show that*

$$I(W) = \min_{Q_{Y|S}} \max_{x \in \mathcal{X}} D(P_{YS|X=x} \| Q_{YS})\,.$$

*Consider an arbitrary sequence of codes at rate $R$ sending $M_n = \lceil 2^{nR} \rceil$ messages over $W^n$. For each $n$, the code is given by an encoder $e_n : [M_n] \to \mathcal{X}^n$ and a decoder $d_n : \mathcal{Y}^n \times \mathcal{S}^n \to [M_n]$. As usual we consider the case*

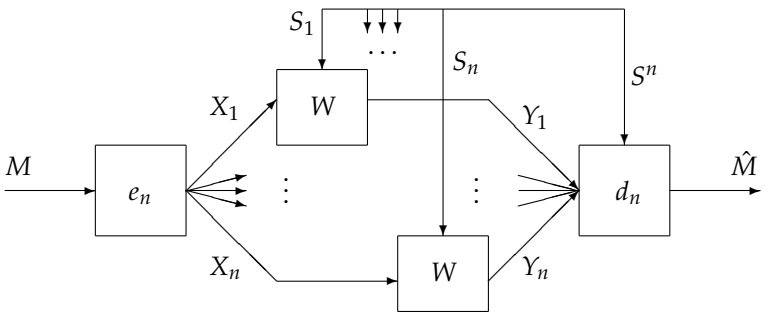



Figure 7.3: The channel coding setup for a fixed blocklength $n \in \mathbb{N}$.

*where $M \in [M_n]$ follows a uniform distribution. Let $\varepsilon_n = P[M \neq \hat{M}]$. We want to show the strong converse, namely that for any such sequence of codes with $\lim_{n\to\infty} \varepsilon_n < 1$, we must have $R \leq I(W)$.*

*Using the meta-converse in the form of Eq. (7.31), we can conclude that for any sequence of codes with $\lim_{n\to\infty} \varepsilon_n < 1$, there exists a $\mu \in (0,1)$ such that, for sufficiently large $n$, we have*

$$nR \leq \max_{x^n \in \mathcal{X}^n} D_s^{1-\mu}(P_{Y^n S^n|X^n=x^n} \| \hat{Q}_{YS}^{\times n}) + \log \frac{2}{\mu},$$

*where $\hat{Q}_{YS}(y,s) = P_S(s)\hat{Q}_{Y|S}(y|s)$ with $\hat{Q}_{Y|S}$ the minimizer of Part c). Thus, in particular, $D(P_{YS|X=x} \| \hat{Q}_{YS}) \leq I(W)$ for all $x \in \mathcal{X}$.*

*d) Show that, for any $\mu \in (0,1)$ and any $\nu > 0$, and sufficiently large $n$,*

$$D_s^{1-\mu}(P_{Y^n S^n|X^n=x^n} \| \hat{Q}_{YS}^{\times n}) \leq n\,(I(W) + \nu)$$

*for all $x^n \in \mathcal{X}^n$. Complete the proof of the strong converse.*

# 8
# *Learning theory: Complexity lower bounds*

**Intended learning outcomes:**

- You understand the concept of sample complexity for distribution learning.
- You know how to show lower bounds on sample complexity by relating it to a channel coding problem.
- You understand how multi-armed bandits are used to investigate exploration vs. exploitation trade-offs.
- You can lower bound the minimax regret by construction adversarial distributions.

**Book reference:** For more information on bandits, see Lattimore & Szepesvári[1].

[1] T. Lattimore and C. Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020

## *8.1 Sample complexity of distribution learning*

### *8.1.1 Problem setup and objective*

Assume the following simple scenario. You are given random samples $X^n = (X_1, X_2, \dots, X_n)$ taken from a DMS $\boldsymbol{X}$ with an unknown distribution $P$ and are asked to provide an estimate $\hat{P}(X^n)$ of $P$ such that

$$\mathbb{E}\left[\delta_{\mathrm{tvd}}\left(P, \hat{P}(X^n)\right)\right] \leq \epsilon\,. \tag{8.1}$$

for some $\epsilon \in (0,1)$. We are interested in the number of samples, $n$, that we need to achieve this. This is called the *sample complexity* for learning (or estimating) the distribution. Generally the sample complexity will depend on $\epsilon$ and on the alphabet size of $P$, which we denote by $d$. Usually we do not care about constants here, but only about how the sample complexity scales as a function of $d$ and $\epsilon$.

**Theorem 8.1.** *The sample complexity of distribution learning is*

$$n = \Theta\left(\frac{d}{\epsilon^2}\right)\,, \tag{8.2}$$

*where $d$ is the alphabet size and $\epsilon$ is defined in (8.1).*

### 8.1.2 Upper bound via an explicit algorithm

Here we need to exhibit a protocol for estimation and analyse it. The protocol is quite obvious — our best estimate is the empirical distribution of the sequence $x^n$ we observe. That is, we choose

$$\hat{P}(x^n) : y \mapsto P_{[x^n]}(y) = \frac{1}{n}\left|\{i \in [n] : x_i = y\}\right|. \tag{8.3}$$

Using this, we can bound

$$\begin{aligned}
\mathbb{E}\left[\delta_{\text{tvd}}\left(P, \hat{P}(X^n)\right)\right] &= \frac{1}{2}\sum_x \mathbb{E}\left[\left|P(x) - P_{[X^n]}(x)\right|\right] && (8.4)\\
&\leq \frac{1}{2}\sum_x \sqrt{\mathbb{E}\left[\left(P(x) - P_{[X^n]}(x)\right)^2\right]} && (8.5)\\
&= \frac{1}{2}\sum_x \sqrt{\text{Var}\left[P_{[X^n]}(x)\right]}, && (8.6)
\end{aligned}$$

where we used that $\mathbb{E}[P_{[X^n]}(x)] = P(x)$. Now note that

$$\begin{aligned}
\text{Var}\left[P_{[X^n]}(x)\right] &= \frac{1}{n^2}\text{Var}\left[\left|\{i \in [n] : X_i = x\}\right|\right] && (8.7)\\
&= \frac{P(x)(1-P(x))}{n} \leq \frac{P(x)}{n}. && (8.8)
\end{aligned}$$

In the above we used that the random variable $\left|\{i \in [n] : X_i = x\}\right|$ follows a binomial distribution. Combing this with (8.6) yields

$$\mathbb{E}\left[\delta_{\text{tvd}}\left(P, \hat{P}(X^n)\right)\right] \leq \frac{1}{2}\sum_x \sqrt{\frac{P(x)}{n}} \leq \frac{1}{2}\sqrt{\frac{d}{n}}. \tag{8.9}$$

Hence, choosing $n = \frac{d}{4\epsilon^2}$ yields the desired accuracy $\epsilon$. Thus, we have shown the upper bound $n = O(d/\epsilon^2)$.

Finally, we note that by Markov's inequality, any protocol satisfying the expectation constraint in (8.1) also produces a close estimate with high probability; namely

$$P\left[\delta_{\text{tvd}}\left(P, \hat{P}(X^n)\right) > \eta\epsilon\right] < \frac{1}{\eta}. \tag{8.10}$$

In fact, much stronger concentration bounds on $\eta$ are possible, but we will not need them here.

### 8.1.3 Lower bound via Fano's inequality

The idea behind this proof is to design a communication scenario for which the learner would be a valid decoder, so that any lower

bounds that we have on the performance of the decoder carry over to the learner. To make decoding difficult, the channel is chosen almost random (with very low mutual information).

For simplicity of exposition we show the lower bound only for even $d$. To do so we first construct a channel with input alphabet $\mathcal{X} = \{x \subset [d] : |x| = d/2\}$ and output alphabet $\mathcal{Y} = [d]$. That is, the inputs are subsets of $[d]$ of size $d/2$. The conditional pmf for $W$ is given by

$$W(y|x) = \frac{1}{d}\left(1 + (-1)^{\mathbf{1}\{y \notin x\}}\mu\right) \tag{8.11}$$

for some $\mu \in [0,1]$ to be specified later. The channel mutual information of $W$ can be bounded using the fact that $I(X : Y) = H(Y) - H(Y|X) \leq \log d - H(Y|X)$ and noting that the upper bound is independent of the input distribution. Hence,

$$\begin{aligned}
I(W) &\leq \log d - H(Y|X) && (8.12)\\
&= \log d + \frac{1+\mu}{2}\log\frac{1+\mu}{d} + \frac{1-\mu}{2}\log\frac{1-\mu}{d} && (8.13)\\
&= \frac{1+\mu}{2}\log(1+\mu) + \frac{1-\mu}{2}\log(1-\mu) && (8.14)\\
&\leq \frac{1+\mu}{2}\mu\log e - \frac{1-\mu}{2}\mu\log e && (8.15)\\
&= \mu^2 \log e\,, && (8.16)
\end{aligned}$$

where we used that $\ln(1+t) \leq t$ for $t > -1$ to establish the second inequality. Hence, the capacity of this channel is very small for small $\mu$ and grows quadratically in $\mu$.

Next we want to construct a (hopefully large) set $\mathcal{A} \subset \mathcal{X}$ such that

$$\delta_{\text{tvd}}(W(\cdot|x), W(\cdot|x')) \geq \frac{\mu}{2} \tag{8.17}$$

for all $x, x' \in \mathcal{A}$ with $x \neq x'$. First, we note that

$$\begin{aligned}
&\delta_{\text{tvd}}(W(\cdot|x), W(\cdot|x')) && (8.18)\\
&= \frac{1}{2}\sum_y \left|\frac{1}{d}\left(1 + (-1)^{\mathbf{1}\{y\notin x\}}\mu\right) - \frac{1}{d}\left(1 + (-1)^{\mathbf{1}\{y\notin x'\}}\mu\right)\right| && (8.19)\\
&= \frac{\mu}{2d}\sum_y \left|(-1)^{\mathbf{1}\{y\notin x\}} - (-1)^{\mathbf{1}\{y\notin x'\}}\right| && (8.20)\\
&= \frac{\mu}{d}\sum_y \mathbf{1}\{y \in x \wedge y \notin x'\} + \mathbf{1}\{y \in x' \wedge y \notin x\} && (8.21)\\
&= \frac{2\mu}{d}\left|x^c \cap x'\right|, && (8.22)
\end{aligned}$$

where in the last step we used that $|x^c \cap x'| = |(x')^c \cap x|$ as both sets are of size $n/2$. Hence, satisfying the condition in (8.17) simply amounts to ensuring that $|x^c \cap x'| \geq \frac{d}{4}$ for all pairs $x, x' \in \mathcal{A}$.

Let us now randomly construct sets in $\mathcal{X}$ as follows. We first sample $d/2$ fair coin tosses $Y_1, Y_2, \ldots, Y_{d/2}$. Then we construct

$$X = \bigcup_{k=1}^{d/2} X_k, \quad \text{where} \quad X_k = \begin{cases} \{k\} & \text{if } Y_k = 0 \\ \{d/2 + k\} & \text{if } Y_k = 1 \end{cases}. \tag{8.23}$$

Now assume that $X$ is drawn as above and $x'$ is fixed but of the same form. Then,

$$P\left[\left|X^c \cap x'\right| < \frac{d}{4}\right] = P\left[\frac{2}{d}\sum_{k=1}^{d/2} \mathbf{1}\{Y_k \neq y'_k\} < \frac{1}{4}\right] \tag{8.24}$$

$$\leq P\left[\left|\frac{2}{d}\sum_{k=1}^{d/2} \mathbf{1}\{Y_k \neq y'_k\} - \frac{1}{2}\right| > \frac{1}{4}\right] \tag{8.25}$$

$$\leq 2e^{-\frac{d}{16}}, \tag{8.26}$$

where we used Hoeffding's inequality (Proposition 0.6) in the last step, leveraging the fact that $\mathbf{1}\{Y_k \neq y'_k\}$ is a Bernoulli random variable with mean $\frac{1}{2}$. We can now construct the set $\mathcal{A}$ recursively. Assuming $A$ already has $\ell$ elements, by the union bound we find that a randomly chosen $X$ satisfies

$$P\left[\forall x' \in \mathcal{A} : \left|X^c \cap x'\right| \geq \frac{d}{4}\right]$$

$$= 1 - P\left[\exists x' \in \mathcal{A} : \left|X^c \cap x'\right| < \frac{d}{4}\right] \geq 1 - 2\ell e^{-\frac{d}{16}}, \tag{8.27}$$

which is positive as long as $\ell < \frac{1}{2}e^{\frac{d}{16}}$. Hence, an $x$ satisfying all these constraints exist and we can add it to the set. This ultimately yields a set $\mathcal{A}$ with $|\mathcal{A}| \geq \frac{1}{2}e^{\frac{d}{16}}$ containing only elements that pairwise satisfy the distance constraint in (8.17).

Using Fano's inequality, we can give a bound on the number of distinct messages $M$ that can be sent through the channel when it is used $n$ times and we require an average probability of error not exceeding $\eta$. We will do this in Exercise 8.1. It yields the bound

$$\log M \leq \frac{nI(W) + 1}{1 - \eta} \tag{8.28}$$

This bound holds for all coding schemes. Now, one particular way to use this channel to send a message in $\mathcal{A}$ is to use a repetition code and transmit the same symbol $x \in \mathcal{A}$ a total of $n$ times through the channel. As a decoder we will use any distribution learning algorithm. Assuming the input is $x$, such an algorithm will give us a distribution $\hat{P}$ that is at most $2\epsilon$ away from $W(\cdot|x)$ with probability at least $\frac{1}{2}$. If we choose $\mu > 8\epsilon$ then we can simply decode to

$$\hat{x} = \operatorname*{argmin}_{x \in \mathcal{A}} \delta_{\text{tvd}}\left(W(\cdot|x), \hat{P}(y^n)\right), \tag{8.29}$$

Verify the details of this argument. Can you see how this can also be used to show the converse to the channel coding theorem?

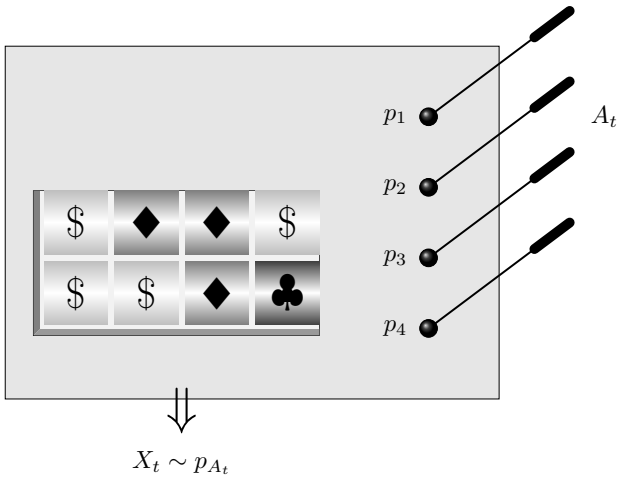



Figure 8.1: A bandit with 4 arms, each inducing a distribution $p_i$ on the reward. Each round $t \in [n]$ an action $A_t$ is chosen and the reward $X_t$ is governed by the distribution $p_{A_t}$.

and since by our choice of $\mu$ any two distinct elements of $\mathcal{A}$ have tvd larger than $4\epsilon$, this achieves an average probability of error at most $\frac{1}{2}$. Plugging these parameter choices into Fano's inequality, we thus must have

$$\frac{d}{16}\log e - 1 \leq \log|\mathcal{A}| < 2(64n\epsilon^2 \log e + 1) \tag{8.30}$$

Solving this for $n$ yields the desired asymptotic behaviour,

$$n > \left(\frac{d}{16}\log e - 3\right)\frac{1}{128\epsilon^2 \log e}, \tag{8.31}$$

which becomes a nontrivial bound for sufficiently large $d$, and shows the desired asymptotic behaviour, $n = \Omega\left(\frac{d}{\epsilon^2}\right)$.

## 8.2 Multi-armed stochastic bandits

### 8.2.1 Problem setup and objective

Multi-armed stochastic bandits are an example of an unsupervised learning problem, where decisions have to made under uncertainty. Bandit models are used to investigate tradeoffs between exploration and exploitation. Exploration here means that we want to learn properties of the various arms (namely their expected rewards) by observing samples so as to find the arm with the highest expected reward. Exploitation means that we mostly want to play the arms which we think have the highest expected rewards. But clearly at the start we do not know yet which arms have higher rewards, so some exploration is necessary before exploitation can occur.

Before we continue let us first formally define the problem.

A *multi-armed stochastic bandit* is given by a set $\mathcal{A}$ that is called the *action set* (which we here assume to be finite). Furthermore, the bandit is in an *environment*, a collection of pdfs $\nu = \{p_a : a \in \mathcal{A}\}$. The environment $\nu \in \mathcal{P}$ is unknown but taken from some (known) set of potential environments $\mathcal{P}$.

At each round $t \in [n]$ a learner chooses an action $a_t \in \mathcal{A}$ and receives a reward $X_t \in \mathbb{R}$ according to the (a priori unknown) pdf $p_{a_t}$.

A *policy* $\pi$ for a multi-armed stochastic bandit is a set of conditional probability distributions

$$\pi_t(a_t|x_1, a_1, \ldots, x_{t-1}, a_{t-1}) \tag{8.32}$$

for $t \in [n]$ from which the player samples the action taken in round $t$ given the previous actions and outcomes.

Given a bandit and a policy we can now discuss the induced joint distribution of rewards and actions. Namely, the joint pdf of $X_1, \ldots X_n$ and $A_1 \ldots A_n$ is given by

$$p(x_1, a_1, \ldots, x_n, a_n) = \prod_{t=1}^{n} \pi_t(a_t|x_1, a_1, \ldots, x_{t-1}, a_{t-1}) p_{a_t}(x_t) \tag{8.33}$$

Based on this, we introduce the following definitions that allow us to quantitatively evaluate policies.

- The *expected reward* of an action $a \in \mathcal{A}$ is defined as $\mu_a = \int p_{a_t}(x)\, x \, \mathrm{d}x$, i.e., the expected reward is $E[X]$ where $X$ is distributed according to the pdf $p_{a_t}$.
- The *maximal expected reward* is defined as $\mu^* = \max_{a \in \mathcal{A}} \mu_a$. It simply corresponds to the reward we expect when we choose the optimal arm.
- The expected *regret* for a policy $\pi$ on a bandit $\nu$ is defined as

$$R_n(\pi, \nu) := n\mu^* - \mathbb{E}\left[\sum_{t=1}^{n} X_t\right], \tag{8.34}$$

  where the expectation is taken over the randomness introduced by the reward distribution and the policy, in case it is non-deterministic.
- The *worst-case regret* of a policy $\pi$ over the environments $\mathcal{P}$ is defined as

$$R_n^*(\pi, \mathcal{P}) := \sup_{\nu \in \mathcal{P}} R_n(\pi, \nu) \tag{8.35}$$

- Finally, the *minimax regret* for $\mathcal{P}$ is defined as

$$R_n^*(\mathcal{P}) := \inf_\pi R_n^*(\pi, \mathcal{P}) = \inf_\pi \sup_{\nu \in \mathcal{P}} R_n(\pi, \nu)\,. \tag{8.36}$$

Our objective is to find how the minimax regret scales with the number of samples $n$ and the number of arms $k$. As we have seen with the other problems we discussed in this module, there are two directions we have to approach this from. On the one hand, we might want to come up with good policies that achieve a small worst-case regret. This is in some sense analogous to the "achievability" problem in source or channel coding, where we also need exhibit a code with the desired properties. The regret for any policy obviously gives an upper bound on the minimax regret. On the other hand, we also want to find lower bounds on the minimax regret that hold for all policies. That is analogous to the "converse" direction in channel or source coding. If the upper and lower bounds match then we know that our policy is optimal.

Show that we have $R_n(\pi, \nu) \geq 0$ for all $\pi$. Which policy $\pi$ achieves the minimal regret for a fixed (and known) $\nu$?

During this lecture we will only be able to derive a lower bound and you will have to believe me that this lower bound is in fact tight. Finding a good policy and analysing it is a bit outside the realm of traditional information theory and more similar to the task of algorithm design and analysis in computer science. If you are interested in this topic, have a look at this book.[2]

[2] T. Lattimore and C. Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020

### 8.2.2 *A lower-bound on minimax regret*

In this section we will show the following theorem:

**Theorem 8.2.** *Let $k \geq 1$, $n \geq k-1$ and let $\mathcal{P}$ be a class of environments with $k$ arms that allows for rewards with Gaussian pdfs $\mathcal{N}(\,\cdot\,; \mu_i, 1)$ for $\mu_i \in [0,1]$ for all $i \in [k]$. Then,*

$$R_n^*(\mathcal{P}) \geq \frac{1}{27}\sqrt{(k-1)n}\,. \tag{8.37}$$

This means that, independent of the policy chosen, the worst-case regret scales at least as the square root of the number of trials and the number of arms, or $R_n^*(\pi, \mathcal{P}) \geq \frac{1}{27}\sqrt{(k-1)n}$ for any policy $\pi$. Note that the lower bound becomes trivial for $k = 1$, which is what we expect, as in this case we have no choice but to take the one arm and our choice is thus always optimal.

To show this, given any $\pi$, we will have to construct at least one environment $\nu$ that exhibits a regret that allows for this lower bound. So, what we will need to show is

$$\forall \pi, \exists \nu : R_n(\pi, \nu) \geq \frac{1}{27}\sqrt{(k-1)n}\,. \tag{8.38}$$

The proof of this statement follows in the next few sections.

### 8.2.3 Decomposing the regret

The first lemma allows us to decompose the regret:

**Lemma 8.3.** *Define $\Delta_a = \mu^* - \mu_a$, called the sub-optimality gap, and let $T_n(a) = \sum_{t=1}^n \mathbf{1}\{A_t = a\}$ be the random variable counting the number of times the action $a \in \mathcal{A}$ is chosen. Then,*

$$R_n(\pi, \nu) = \sum_{a \in \mathcal{A}} \Delta_a \, \mathbb{E}[T_n(a)] \,. \tag{8.39}$$

Note that the distribution of $T_n(a)$ depends on both the environment and the policy.

*Proof.* Starting from the definition of the regret and the fact that $\sum_{a \in \mathcal{A}} \mathbb{E}[T_n(a)] = n$ and $\sum_{a \in \mathcal{A}} \mathbf{1}\{A_t = a\} = 1$, we have

$$\begin{aligned} R_n(\pi, \nu) &:= n\mu^* - \mathbb{E}\left[\sum_{t=1}^n X_t\right] && (8.40)\\ &= \sum_{a \in \mathcal{A}} \mathbb{E}[T_n(a)]\mu^* - \mathbb{E}\left[\sum_{t=1}^n \sum_{a \in \mathcal{A}} \mathbf{1}\{A_t = a\} X_t\right] && (8.41)\\ &= \sum_{a \in \mathcal{A}} \left( \mathbb{E}[T_n(a)]\mu^* - \mathbb{E}\left[\sum_{t=1}^n \mathbf{1}\{A_t = a\} X_t\right] \right) . && (8.42) \end{aligned}$$

Now we observe that in the latter expectation the action $a$ is fixed and thus the random variables $X_t$ are drawn independently from $p_a$. The expectation value thus factorises to

$$\mathbb{E}\left[\sum_{t=1}^n \mathbf{1}\{A_t = a\} X_t\right] = \mathbb{E}\left[\sum_{t=1}^n \mathbf{1}\{A_t = a\}\right] \mu_a = \mathbb{E}[T_n(a)]\mu_a, \tag{8.43}$$

which implies the desired result when plugged into (8.42) □

### 8.2.4 Constructing worst-case environments

Without loss of generality we can take the action set to be $\mathcal{A} = [k]$. Let us introduce a vector of means $\mu \in [0,1]^k$ and then the corresponding environment as

$$v_\mu = \{\mathcal{N}(\,\cdot\,; \mu_1, 1), \mathcal{N}(\,\cdot\,; \mu_2, 1), \ \ldots\ , \mathcal{N}(\,\cdot\,; \mu_k, 1)\} \,. \tag{8.44}$$

To show the lower bound in Theorem 8.2, for every policy $\pi$, it is sufficient to find two vectors $\mu$ and $\mu'$ such that either $R_n(\pi, v_\mu)$ or $R_n(\pi, v_{\mu'})\}$ exceeds $\frac{1}{27}\sqrt{(k-1)n}$, since that implies that the maximum of the two does too. Moreover, to ensure this, it is sufficient to show that

$$R_n(\pi, v_\mu) + R_n(\pi, v_{\mu'}) \geq \frac{2}{27}\sqrt{(k-1)n} \,. \tag{8.45}$$

We will thus next show this inequality by constructing such vectors for a fixed policy $\pi$. We choose $\mu = (\Delta, 0, \dots, 0)$ for some $\Delta \in (0, 1]$ still to be specified. Clearly the optimal policy for $v_\mu$ would always choose the first action. Now consider a run of the algorithm with policy $\pi$ on environment $v_\mu$. Then we define

$$i_* := \operatorname*{argmin}_{i \in [k] \setminus \{1\}} \{\mathbb{E}[T_n(i)]\}, \tag{8.46}$$

the action that is chosen the least number of times (in expectation) by the policy $\pi$ when run on the environment $v_\mu$. Since $\sum_{i=1}^k \mathbb{E}[T_n(i)] = n$ we must have

$$\mathbb{E}[T_n(i_*)] \leq \frac{n}{k-1}. \tag{8.47}$$

by the pigeonhole principle. Indeed, if on the contrary $\mathbb{E}[T_n(i_*)] > \frac{n}{k-1}$, then we get a contradiction since

$$n = \sum_{i=1} \mathbb{E}[T_n(i)] \geq \sum_{i=2} \mathbb{E}[T_n(i)] \geq (k-1)\mathbb{E}[T_n(i_*)] \tag{8.48}$$

$$> (k-1)\frac{n}{k-1} = n. \tag{8.49}$$

We can now define our alternative environment as

$$\mu' = (\Delta, 0, \dots, 0, \underbrace{2\Delta}_{\text{at position } i_*}, 0, \dots, 0). \tag{8.50}$$

Note that this a worst-case scenario in the sense that if we run policy $\pi$ and think that the environment is $\mu$ instead of $\mu'$ then our regret would be maximal as we play $i_*$ the least often. Obviously our policy should be clever enough so that we at some point learn that the environment is $\mu'$ and adapt our action choices accordingly, but as we will see by choosing $\Delta$ small we can make it very difficult to distinguish the two cases.

According to Lemma 8.3 we can decompose and then bound the two regrets as

$$R_n(\pi, v_\mu) = \Delta \sum_{i \in [k] \setminus \{1\}} \mathbb{E}[T_n(i)] = \Delta\left(n - \mathbb{E}[T_n(1)]\right), \tag{8.51}$$

$$R_n(\pi, v_\mu) = \Delta\mathbb{E}'[T_n(1)] + 2\Delta \sum_{i \in [k] \setminus \{1, i_*\}} \mathbb{E}'[T_n(i)] \geq \Delta\mathbb{E}'[T_n(1)]. \tag{8.52}$$

Here we used $\mathbb{E}'$ to denote the expectation under the distribution $p'$ induced by $\pi$ and $v_{\mu'}$. We can further bound

$$\mathbb{E}[T_n(1)] \leq \frac{n}{2} p\left[T_n(1) < \frac{n}{2}\right] + np\left[T_n(1) \geq \frac{n}{2}\right] \tag{8.53}$$

$$= \frac{n}{2}\left(1 + p\left[T_n(1) \geq \frac{n}{2}\right]\right) \tag{8.54}$$

and, using Markov's inequality, $\mathbb{E}'[T_n(1)] \geq \frac{n}{2} p'\left[T_n(1) \geq \frac{n}{2}\right]$. This yields

$$R_n(\pi, v_\mu) \geq \frac{n\Delta}{2}\left(1 - p\left[T_n(1) \geq \frac{n}{2}\right]\right) = \frac{n\Delta}{2} p\left[T_n(1) < \frac{n}{2}\right] \tag{8.55}$$

$$R_n(\pi, v_{\mu'}) \geq \frac{n\Delta}{2} p'\left[T_n(1) \geq \frac{n}{2}\right]. \tag{8.56}$$

And, thus,

$$R_n(\pi, v_\mu) + R_n(\pi, v_{\mu'}) \geq \frac{n\Delta}{2}\left(p\left[T_n(1) < \frac{n}{2}\right] + p'\left[T_n(1) \geq \frac{n}{2}\right]\right). \tag{8.57}$$

### 8.2.5 *Lower-bounding the regret*

This is really where the information theory tools come in! The next lemma gives us a lower bound on the sum of the probabilities of two complementary events, evaluated on two (generally different) distributions. When these distributions are similar we expect that the sum of probabilities is close to 1.

**Lemma 8.4** (Bretagnolle-Huber inequality)**.** *Let $p$ and $q$ be two pdfs for the same random variable $X$ taking values on $\mathcal{X}$. For any $A \subset \mathcal{X}$, we have*

$$p(A) + q(A^c) \geq \frac{1}{2}\exp\left(-D(p\|q)\right), \tag{8.58}$$

*where $D(p\|q) = \int_{\mathcal{X}} p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x$.*

*Proof.* We first note that

$$p(A) + q(A^c) = \int_A p(x)\mathrm{d}x + \int_{A^c} q(x)\mathrm{d}x \geq \int_{\mathcal{X}} \min\{p(x), q(x)\}\mathrm{d}x. \tag{8.59}$$

Furthermore, using the Cauchy-Schwartz inequality, we find

$$\begin{aligned}
&\left(\int_{\mathcal{X}} \sqrt{p(x)q(x)}\,\mathrm{d}x\right)^2 \\
&\quad = \left(\int_{\mathcal{X}} \sqrt{\min\{p(x), q(x)\}\max\{p(x), q(x)\}}\,\mathrm{d}x\right)^2 && (8.60)\\
&\quad \leq \left(\int_{\mathcal{X}} \min\{p(x), q(x)\}\mathrm{d}x\right)\left(\int_{\mathcal{X}} \max\{p(x), q(x)\}\mathrm{d}x\right) && (8.61)\\
&\quad \leq 2\int_{\mathcal{X}} \min\{p(x), q(x)\}\mathrm{d}x. && (8.62)
\end{aligned}$$

Thus, combining this with Eq. (8.59), we have

$$p(A) + q(A^c) \geq \frac{1}{2}\exp\left(2\log\int_{\mathcal{X}} p(x)\sqrt{\frac{q(x)}{p(x)}}\,\mathrm{d}x\right). \tag{8.63}$$

Finally, using Jensen's inequality for the logarithm, we arrive at

$$p(A) + q(A^c) \geq \frac{1}{2} \exp\left(2 \int_{\mathcal{X}} p(x) \log \sqrt{\frac{q(x)}{p(x)}} \, \mathrm{d}x\right) \tag{8.64}$$

$$= \frac{1}{2} \exp\left(-\int_{\mathcal{X}} p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x\right) \tag{8.65}$$

$$= \frac{1}{2} \exp\left(-D(p\|q)\right) . \tag{8.66}$$

□

We can now continue with our derivation of a lower bound. Applying this Lemma to our bound in Eq. (8.57), we find

$$R_n(\pi, v_\mu) + R_n(\pi, v_{\mu'}) \geq \frac{n\Delta}{4} \exp\left(-D(p\|p')\right) , \tag{8.67}$$

and it then remains to evaluate this relative entropy for our two distributions.

**Lemma 8.5** (Divergence decomposition lemma)**.** *Let $\pi$ be any policy and $p$ and $p'$ be the distributions induce by the environments $(p_1, p_2, \ldots, p_k)$ and $(p'_1, p'_2, \ldots, p'_k)$, respectively. Then,*

$$D(p\|p') = \sum_{i=1}^{k} \mathbb{E}[T_n(i)] D(p_i \| \, p'_i) . \tag{8.68}$$

*Proof.* We first recall that $p(x_1, a_1, \ldots, x_n, a_n)$ decomposes into reward and action probabilities as $\prod_{t=1}^{n} \pi_t(a_t|x_1, a_1, \ldots, x_{t-1}, a_{t-1}) p_{a_t}(x_t)$. As a consequence, we can simplify

$$D(p\|p') = \mathbb{E}\left[\log \frac{p(X_1, A_1, \ldots, X_n, A_n)}{p'(X_1, A_1, \ldots, X_n, A_n)}\right] \tag{8.69}$$

$$= \mathbb{E}\left[\sum_{t=1}^{n} \log \frac{p_{A_t}(X_t)}{p'_{A_t}(X_t)}\right] \tag{8.70}$$

$$= \sum_{t=1}^{n} \sum_{i=1}^{k} P[A_t = i] \, \mathbb{E}\left[\log \frac{p_{A_t}(X_t)}{p'_{A_t}(X_t)} \middle| A_t = i\right] \tag{8.71}$$

$$= \sum_{i=1}^{k} \mathbb{E}\left[\sum_{t=1}^{n} \delta_{A_t,i}\right] \, \mathbb{E}\left[\log \frac{p_i(X_t)}{p'_i(X_t)} \middle| A_t = i\right] \tag{8.72}$$

$$= \sum_{i=1}^{k} \mathbb{E}[T_n(i)] D(p_i \| p'_i) , \tag{8.73}$$

where we used the law of total expectation to get the third equality and in the penultimate step we used that $X_t$ is drawn independently from $p_i$ once $A_t = i$ is fixed. □

Applying this to our particular situation we find that

$$D(p\|p') = \sum_{i=1}^{k} \mathbb{E}[T_n(i)]\, D\left(\mathcal{N}(\,\cdot\,;\mu_i,1)\|\mathcal{N}(\,\cdot\,;\mu_i',1)\right) \tag{8.74}$$

$$= \mathbb{E}[T_n(i_*)]\,\frac{(2\Delta)^2}{2} \tag{8.75}$$

$$\leq \frac{2\Delta^2 n}{k-1}\,, \tag{8.76}$$

where we realised that $\mu_i$ and $\mu_i'$ only differ at $i = i_*$ and evaluated the relative entropy for two normal Gaussian distributions with different means. In the last step we used Eq. (8.47).

Substituting this into Eq. (8.67), we find

$$R_n(\pi, v_\mu) + R_n(\pi, v_{\mu'}) \geq \frac{n\Delta}{4}\exp\left(-\frac{2\Delta^2 n}{k-1}\right). \tag{8.77}$$

Verify that

$$D\left(\mathcal{N}(\,\cdot\,;\mu,1)\|\mathcal{N}(\,\cdot\,;\mu',1)\right) = \frac{(\mu-\mu')^2}{2}\,.$$

Choosing now $\Delta = \sqrt{\frac{k-1}{4n}}$ yields

$$R_n(\pi, v_\mu) + R_n(\pi, v_{\mu'}) \geq \sqrt{n(k-1)}\,\underbrace{\frac{1}{8}\exp\left(-\frac{1}{2}\right)}_{\geq \frac{2}{27}}, \tag{8.78}$$

which is what we set out to show.

## 8.3 Exercises

**Exercise 8.1.** *Suppose one can send $M$ distinct messages through $n$-copies of a memoryless channel $W$ with average error at most $\eta$. Use Fano's inequality to show that*

$$\log M \leq \frac{nI(W)+1}{1-\eta}.$$

**Exercise 8.2.** *In this exercise, we will derive a sample complexity for distribution learning with respect to $\ell_2$-norm: instead of requiring $\delta_{tvd}(P, \hat{P}_{X^n}) \leq \epsilon$, we would like to guarantee $\|P - \hat{P}_{X^n}\|_2 \leq \epsilon$. We are also going to explicitly include the scaling in the probability of error, $\eta$, in our bound.*

1. *Show that the empirical estimator achieves $P[\|P - \hat{P}_{X^n}\|_2 \leq \epsilon] \geq 1 - \eta$ with*

$$n = O\left(\frac{d}{\epsilon^2}\log\left(\frac{d}{\eta}\right)\right) \tag{8.79}$$

2. *It turns out that the above bound on the sample complexity is not tight. In fact, it can be shown that learning in $l_2$ is possible using only $n = O(\log(1/\eta)/\epsilon^2)$ samples - without dependence on the dimension of the distribution, a stark contrast to learning in $\delta_{tvd}$! Assume such an algorithm is given: argue that the scaling in $\epsilon$, $O(1/\epsilon^2)$, of this algorithm has to be tight. Assume that there exists a more efficient learner using only $O(1/\epsilon^\alpha)$ samples, for $\alpha < 2$ and find a contradiction.*

If you are interested in more details, see arXiv:2002.11457. The same paper also proves the tight sample complexity for learning in variation distance including $\eta$, which is in $\Theta([d + \log(1/\eta)]/\epsilon^2)$.

**Hint:** Note that the approach from the lecture would result in $O(d/\epsilon^2\eta)$. Instead, you may ensure that none of the events $L(x) := \mathbf{1}\{|P(x) - \hat{P}_{X^n}(x)| \geq \epsilon/\sqrt{d}\}$ take place and then apply a Hoeffding bound, i.e. for independent random variables $X_1 \cdots X_n$ satisfying $X_i \in [a_i, b_i]$ and $S_n = \sum_{i=1}^n X_i$

$$P[S_n - \mathbb{E}[S_n] \geq t] \leq e^{-\frac{t^2}{\sum_{i=1}^n (b_i - a_i)^2}}$$

**Hint:** In the lecture you have seen that for $u, v \in \mathbb{R}^d$, $\|u \cdot v\|_1 \leq \|u\|_2\|v\|_2$. How do you have to choose $u$ and $v$ to obtain a relation between $\delta_{\text{tvd}}(P, \hat{P}_{X^n})$ and $\|P - \hat{P}_{X^n}\|_2$?

**Exercise 8.3.** *Consider the same setup as in Theorem 8.2 but replacing $\mathcal{P}$ by the class of environments with k arms that allows for rewards with Gaussian bounded variance, i.e if $\nu \in \mathcal{P}$ then*

$$\nu = \{\mathcal{N}\left(\cdot; \mu_a, \sigma_a^2\right) : a \in [k], \mu_a \in [0,1], \sigma_{\min}^2 \leq \sigma_a^2 \leq \sigma_{\max}^2\},$$

*where $0 \leq \sigma_{\min} \leq \sigma_{\max}$.*

1. *Show that*

$$R_n^*(\mathcal{P}) \geq \frac{\sigma_{\max}}{27}\sqrt{kn}.$$

2. *Let $\sigma_{\max}^2 = n'$ for some fixed $n' \in \mathbb{N}$. What is the behaviour of $R_n^*(\mathcal{P})$ for $n \leq n'$? Does it get worse or better the scaling of the minimax regret on the time horizon n? What happens to $R_n^*(\mathcal{P})$ if we allow $\sigma_{\max}^2 \approx 0$?*

**Hint:** You can follow the thread in the lecture notes, without assuming the variance to be 1.

*Discuss your ideas at a high-level understanding the role that play the variance when learning from distributions.*

**Exercise 8.4.** *In this exercise we are going to derive a minimax regret lower bound for a two-armed bandit with Bernouli rewards. Let's denote $\text{Bern}(;\mu)$ the pdf of a Bernoulli distribution with support in $\{0,1\}$ and mean $\mu \in [0,1]$ i.e*

$$\text{Bern}(x;\mu) = \begin{cases} \mu & \text{if } x = 1 \\ 1-\mu & \text{if } x = 0 \\ 0 & \text{else .} \end{cases}$$

*The construction will be very similar to the one from the lecture notes and it involves the computation of a relative entropy between Bernoulli distributions. In order to ease this step we are going to use a "simpler" quantity that is the $\chi^2$-distance which is defined as*

$$\chi^2(p,q) := \int_{\mathcal{X}} \frac{p^2(x)}{q(x)} dx - 1,$$

*for two pdfs p and q of the same random variable X taking values on $\mathcal{X}$.*

1. *Show the following inequality between the relative entropy and the $\chi^2$-distance*

$$D(p\|q) \leq \chi^2(p,q).$$

**Hint:** Use the inequality $\log(1+x) \leq x$ for $x > -1$.

2. *Show that the $\chi^2$-distance for two Bernoulli's with means $\mu, \mu'$ is given by*

$$\chi^2(\text{Bern}(\cdot;\mu), \text{Bern}(\cdot;\mu')) = \frac{(\mu-\mu')^2}{\mu'(1-\mu')}.$$

3. *Consider now $\mu \in [0,1]^2$ and the two-armed Bernoulli environment $\nu_\mu$ given by*

$$\nu_\mu = \{\text{Bern}(\cdot;\mu_a) : a \in [2]\} \quad \textit{where } \mu = \left(\frac{1}{2} + \frac{\Delta}{4}, \frac{1}{2}\right)$$

*for some constant $\Delta \in (0,1]$ to be defined later. Analogously consider $\nu_{\mu'}$ with*

$$\mu' = \left(\frac{1}{2} + \frac{\Delta}{4}, \frac{1}{2} + \frac{\Delta}{2}\right).$$

*Given a policy $\pi$ prove that*

$$R_n\left(\pi, \nu_\mu\right) + R_n\left(\pi, \nu_{\mu'}\right) \geq \frac{n\Delta}{16} \exp\left(-D(p\|p')\right),$$

*where $p, p'$ are the probability distributions of rewards and actions induced by the policy $\pi$ on $\nu_\mu, \nu_{\mu'}$ respectively.*

4. *Use a), b) and the trivial bound $\mathbb{E}\left[T_a(n)\right] \leq n$ to show*

$$D(p\|p') \leq \frac{n\Delta^2}{2(1-\Delta^2)}.$$

5. *Combine all the above results and show that for the class of environments with 2 arms and Bernoulli rewards we can bound the minimax regret as*

$$R_n^*(\mathcal{P}) \geq \frac{1}{864}\sqrt{n+1}$$

**Hint:** Solve $n = \frac{1-\Delta^2}{\Delta^2}$ and use $n \geq \frac{n+1}{2}$ for $n \geq 1$.

**Exercise 8.5.** *In this exercise we will see a slightly different technique to derive lower bounds for problems in learning theory, which will allow us to find lower bounds for a yes/no question regarding distributions: instead of learning all parameters of a distribution, one might often only be interested in testing for a specific property. For example, consider the problem of uniformity testing: given samples $X^n$ from a distribution, decide whether the underlying distribution is uniform or $\epsilon$ far from being uniform.*

*We start with the following lemma, which formalizes the intuitive notion that if a variable $A$ carries little information about a fair random bit $X$, then any attempt to guess $X$ based on observing $A$ can only be marginally better than random.*

***Lemma.*** Let $X$ be a fair random bit, $A$ a random variable correlated with $X$. Then there exist small positive constants $c_1, c_2$ independent of $A$ as follows: if $I(X:A) \leq c_1$, then for any function $f$ trying to recover $X$ from $A$, we have $P[f(A) = X] \leq 1/2 + c_2$.

1. *Prove the lemma. You do not need to determine exact values for $c_1, c_2$, just argue that they are strictly larger than zero independent of $A$. Use that the binary entropy $h_2(p)$ has a single maximum at $p = 1/2$.*

**Hint:** It might be more intuitive to show the contrapositive: if there exists $f$ such that $P[f(A) = X] > 1/2 + c_2$, then $I(A:X) > c_1$. Recall that $I(X:Y) := H(X) - H(X|Y)$, and start from writing out the definition of $H(X|f(A))$. Finally use the data processing inequality, $I(X:f(Y)) \leq I(X:Y)$.

2. *How can we in principle use this lemma to derive lower bounds for a yes/no question about a distribution?*

**Hint:** If $A^n$ is a set of $n$ samples, interpret $I(X : A^n)$ as a function increasing in $n$, and note that we would like to solve the problem reliably, that is with probability close to 1.

3. *For $X$ again a fair random bit and $A$ an arbitrary random variable taking values in an alphabet $S_A$, show that*

$$I(X : A) \leq \frac{\log(e)}{2} \sum_{a \in S_A} \frac{(P[A = a|X = 0] - P[A = a|X = 1])^2}{P[A = a|X = 0] + P[A = a|X = 1]}. \tag{8.80}$$

**Hint:** Use $\ln(x) \leq x - 1$ (or $\frac{1}{\log(e)} D(P\|Q) \leq \chi^2(P\|Q)$) and the fact that $X$ is fair.

4. *Using the above lemma, show a lower bound of $\Omega(1/\epsilon^2)$ for testing whether a coin is fair ($P[\text{Heads}] = 0.5$) or biased by at least $\epsilon$ ($|P[\text{Heads}] - 0.5| \geq \epsilon$). We are only interested in the scaling on $\epsilon$.*

**Hint:** Depending on the input, the output distribution of your channel should correspond to either a fair or a biased coin. When applying the result from c.), use that for $x, y > 0$, $(x-y)^2/(x+y) < (x-y)^2/x = x - 2y + y^2/x$, and simplify further using the binomial formula.

# *Bibliography*


T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991. ISBN 9780471748823. DOI: 10.1002/047174882X.

Imre Csiszár. The Method of Types. *IEEE Transactions on Information Theory*, 44(6):2505–2523, oct 1998. DOI: 10.1109/18.720546.

T. S. Han. *Information-Spectrum Methods in Information Theory*. Applications of Mathematics. Springer, 2002.

Masahito Hayashi and Hiroshi Nagaoka. General Formulas for Capacity of Classical-Quantum Channels. *IEEE Transactions on Information Theory*, 49(7):1753–1768, jul 2003. DOI: 10.1109/TIT.2003.813556.

David A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952. DOI: 10.1109/JRPROC.1952.273898.

R. Impagliazzo, L. A. Levin, and M. Luby. Pseudo-random generation from one-way functions. In *Proc. ACM STOC 1989*, pages 12–24. ACM Press, 1989. ISBN 0897913078. DOI: 10.1145/73007.73009.

T. Lattimore and C. Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Yury Polyanskiy, H. Vincent Poor, and Sergio Verdú. Channel Coding Rate in the Finite Blocklength Regime. *IEEE Transactions on Information Theory*, 56(5):2307–2359, may 2010. DOI: 10.1109/TIT.2010.2043769.

A. Rényi. On Measures of Information and Entropy. In *Proc. 4th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 547–561, Berkeley, California, USA, 1961. University of California Press.

C. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 1948. DOI: 10.1002/j.1538-7305.1948.tb00917.x.

Maurice Sion. On General Minimax Theorems. *Pacific Journal of Mathematics*, 8:171–176, 1958.

Marco Tomamichel and Vincent Y. F. Tan. A Tight Upper Bound for the Third-Order Asymptotics for Most Discrete Memoryless Channels. *IEEE Transactions on Information Theory*, 59(11):7041–7051, nov 2013. DOI: 10.1109/TIT.2013.2276077.