The Mathematical Theory of Information, and Applications (Version 2.0)

Ronald Cramer* and Serge Fehr†

Abstract

These lecture notes introduce some basic concepts from Shannon's *information theory*, such as (conditional) Shannon entropy, mutual information, and Rényi entropy, as well as a number of basic results involving these notions.

Subsequently, well-known bounds on perfectly secure encryption, source coding (i.e. data compression), and reliable communication over unreliable channels are discussed. We also cover and prove the elegant *privacy amplification theorem*. This provides a means to mod out the adversary's partial information and to distill a highly secret key. It is a key result in theoretical cryptography, and a primary starting point for the very active subarea of unconditional security.

1 Introduction

We define several measures on the amount of randomness inherent to a random variable. Or, put in another way, on the amount of uncertainty about the outcome of a process that involves randomness. Depending on the application, one measure may be more adequate or more useful than another.

Our first goal is to define *Shannon entropy* and to prove some elementary properties of it, and to develop the theory a bit further so as to include the concepts of *conditional entropy*, *mutual information* and *conditional mutual information*, and prove some basic properties of these notions. Based on these elementary properties, we can then easily state and prove Shannon's pessimistic result about perfectly secure encryption, which basically says that for perfectly secure encryption one needs a key that is as least as long as the plaintext.

We then proceed to discuss two questions that are of fundamental importance to the theory and practice of information. How much can information be compressed? And, how much information can be reliably sent over an unreliable channel? We show that for both questions, it is the Shannon entropy (and its related measures) that provide the right answers.

The last part of this note is dedicated to the *privacy amplification theorem*, a result that is fundamental to the theory and practice of modern cryptography. In a nut shell, privacy amplification enables to transform a weakly secure key X, about which an adversary has some partial information, into a highly secure key K, about which the adversary has (essentially) no information at all. What is surprising and makes privacy amplification very powerful is the fact that the extraction of the secure key K from the weak key X works *universally*, independently of what kind of information the adversary holds on X, as long as the amount of information he holds on X is bounded with respect to some suitable information measure (which turns out to be the so-called *Rényi entropy*).

More information, motivation and background on most of the topics treated in this note can be found in the book by Cover and Thomas: *Elements of Information Theory* [2], the survey article by Wolf [4], and the IEEE-IT article *Generalized Privacy Amplification* by Bennett, Brassard, Crepeau and Maurer [1].

*CWI, Amsterdam, and Mathematical Institute, Leiden University. `www.cwi.nl/~cramer`.
†CWI, Amsterdam. `www.cwi.nl/~fehr`.

2 Preliminaries

$\mathbb{N}$ denotes the set of positive integers (excluding 0), and $\mathbb{N}_0$ denotes the set of non-negative integers, i.e., $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. $\mathbb{R}$ denotes the set of real numbers, $\mathbb{R}_{>0}$ denotes the set of positive real numbers, and $\mathbb{R}_{\geq 0}$ denotes the set of non-negative real numbers.

The notation $\log(\cdot)$ refers to the binary logarithm function, i.e., the logarithm function with base equal to 2, whereas the notation $\ln(\cdot)$ refers to the natural logarithm function.

2.1 Probabilities and Random Variables

A (finite) *probability space* $(\mathcal{U}, P)$ consists of a finite, non-empty *sample space* $\mathcal{U}$ and a *probability measure* P, which is a function $P : \mathcal{U} \to \mathbb{R}_{\geq 0}$ that satisfies

$$\sum_{u \in \mathcal{U}} P(u) = 1 \,.$$

An *event* $\mathcal{A}$ is a subset $\mathcal{A}$ of the sample space $\mathcal{U}$. Its probability is defined as

$$P[\mathcal{A}] = \sum_{u \in \mathcal{A}} P(u),$$

where by convention $P[\emptyset] = 0$. If $\mathcal{B}$ is another event, we use the notation $P[\mathcal{A}, \mathcal{B}]$ for $P[\mathcal{A} \cap \mathcal{B}]$. For events $\mathcal{A}$ and $\mathcal{B}$ with $P[\mathcal{A}] > 0$, the conditional probability of $\mathcal{B}$ given $\mathcal{A}$ is defined as

$$P[\mathcal{B}|\mathcal{A}] = \frac{P[\mathcal{A}, \mathcal{B}]}{P[\mathcal{A}]} \,.$$

Let $(\mathcal{U}, P)$ be a fixed probability space. A *random variable* X is a function $X : \mathcal{U} \to \mathcal{X}$, where we may assume $\mathcal{X}$ to be finite. The *image* and the *range* of a random variable X are given by the image and the range of X in the function-theoretic sense. For $x \in \mathcal{X}$, let $X = x$ denote the event $\{u \in \mathcal{U} : X(u) = x\}$. The *(probability) distribution* of X is the function $P_X : \mathcal{X} \to [0, 1]$ defined as

$$P_X(x) = P[X = x] \,.$$

A *real* random variable is one whose image is contained in $\mathbb{R}$. If $\mathcal{A}$ is an event with $P[\mathcal{A}] > 0$, then the *conditional* probability distribution of X given $\mathcal{A}$ is given by

$$P_{X|\mathcal{A}}(x) = \frac{P[X = x, \mathcal{A}]}{P[\mathcal{A}]},$$

Note that both $(\mathcal{X}, P_X)$ and $(\mathcal{X}, P_{X|\mathcal{A}})$ themselves form probability spaces.

If X and Y are two random variables defined on the same probability space, with respective ranges $\mathcal{X}$ and $\mathcal{Y}$, then the pair XY is a random variable with probability distribution $P_{XY} : \mathcal{X} \times \mathcal{Y} \to [0, 1]$ given by

$$P_{XY}(x, y) = P[X = x, Y = y] \,.$$

We call P_{XY} the *joint* distribution of X and Y. This naturally extends to three and more random variables. If $P_{XY} = P_X \cdot P_Y$, in the sense that $P_{XY}(x, y) = P_X(x) P_Y(y)$ for all $\in \mathcal{X}$ and $y \in \mathcal{Y}$, then the random variables X and Y are *independent*. If $P_Y(y) > 0$, we often use the notation $P_{X|Y}(\cdot|\cdot)$ defined by

$$P_{X|Y}(x|y) = P_{X|Y=y}(x),$$

where $x \in \mathcal{X}$.

If $f : \mathcal{X} \to \mathcal{Y}$ is a surjective function, then the random variable $f(X)$ is defined by composing the map f with the map X. Its image is $\mathcal{Y}$. Clearly, writing Y for $f(X)$,

$$P_Y(y) = \sum_{x \in \mathcal{X} : f(x) = y} P_X(x).$$

For example, $1/P_X(X)$ denotes the real random variable obtained from X by composing with the map $1/P_X$ that assigns $1/P_X(x) \in \mathbb{R}$ to $x \in \mathcal{X}$.

Let X be a real random variable. Then, the *expectation* of X is defined as

$$E[X] = \sum_{x \in \mathcal{X}} P_X(x) \cdot x.$$

Hoeffding's inequality (here stated for *binary* random variables) states that for a list of independent and identically distributed random variables, the average of the random variables is close to the expectation, except with very small probability.

Proposition 1 (Hoeffding's inequality) *Let $X_1, \ldots, X_n$ be independent and identically distributed binary random variables with $P_{X_i}(0) = 1 - \mu$ and $P_{X_i}(1) = \mu$, and thus $E[X_i] = \mu$. Then, for any $\delta > 0$*

$$P\big[\textstyle\sum_i X_i > (\mu + \delta) \cdot n\big] \leq \exp(-2\delta^2 n)\,.$$

In the remainder of these notes, when we speak of a random variable, we leave the underlying probability space on which it is defined implicit. Unless otherwise stated, a collection of random variables is defined on the same (implicit) probability space, so that their joint distribution is always well-defined.

2.2 Jensen's Inequality

In the following, let $\mathcal{D}$ be an interval in $\mathbb{R}$ and let $f : \mathcal{D} \to \mathbb{R}$ be a real-valued function on $\mathcal{D}$.

Definition 1 *The function $f : \mathcal{D} \to \mathbb{R}$ is* convex *if for all $x_1, x_2 \in \mathcal{D}$ and for all $\lambda \in [0,1] \subset \mathbb{R}$:*

$$\lambda f(x_1) + (1-\lambda) f(x_2) \geq f(\lambda x_1 + (1-\lambda) x_2).$$

The function f is strictly convex *if equality only holds when $\lambda = 0$ or $\lambda = 1$, or when $x_1 = x_2$.*
The function f is (strictly) concave *if the function $-f$ is (strictly) convex.*

In other words "chords lie above the graph of the function."

Lemma 1 *Suppose $\mathcal{D}$ is open and the function $f : \mathcal{D} \to \mathbb{R}$ is such that for all $x \in \mathcal{D}$ the second order derivative $f''(x)$ exists and is non-negative (positive). Then f is convex (strictly convex).*

Proof. This follows directly from the Taylor-series expansion of f in a neighborhood of x_0. Let $x_0, x \in \mathcal{D}$. Then, for some x^* between x_0 and x,

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

Since $f''(x^*) \geq 0$ by assumption, we have, for all $x, x_0 \in (a, b)$,

$$f(x) \geq f(x_0) + f'(x_0)(x - x_0).$$

Now let $x_1, x_2 \in \mathcal{D}$, and let $\lambda \in [0,1]$, as in the definition of convexity. If we set

$$x_0 = \lambda x_1 + (1-\lambda) x_2 \quad \text{and} \quad x = x_1,$$

it follows that

$$f(x_1) \geq f(x_0) + f'(x_0)[(1-\lambda)(x_1 - x_2)].$$

Keeping x_0 as above but setting $x = x_2$ instead, gives

$$f(x_2) \geq f(x_0) + f'(x_0)[\lambda(x_2 - x_1)].$$

The claim now follows by multiplying the first inequality by λ, the second by $1 - \lambda$, and by adding up the results (and writing $\lambda x_1 + (1-\lambda) x_2$ for x_0). □

Examples of strictly convex functions on $\mathbb{R}$ are $f(x) = x^2$ and $f(x) = e^x$. The functions $f(x) = \log x$ and $f(x) = \sqrt{x}$ are examples of strictly concave functions on their respective domains $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$.

Proposition 2 *(**Jensen's inequality**) Let the function $f : \mathcal{D} \to \mathbb{R}$ be convex, and let $n \in \mathbb{N}$. Then for any $p_1, \ldots, p_n \in \mathbb{R}_{\geq 0}$ such that $\sum_{i=1}^n p_i = 1$ and for any $x_1, \ldots, x_n \in \mathcal{D}$ it holds that*

$$\sum_{i=1}^n p_i f(x_i) \geq f\Big(\sum_{i=1}^n p_i x_i\Big).$$

If f is strictly convex and $p_1, \ldots, p_n > 0$, then equality holds iff $x_1 = \cdots = x_n$.

In particular, if X is a real random variable whose image $\mathcal{X}$ is contained in $\mathcal{D}$, then

$$E[f(X)] \geq f(E[X]),$$

and, if f is strictly convex, equality holds iff there is $c \in \mathcal{X}$ such that $X = c$ with probability 1.

Proof. The proof is by induction. The case $n = 1$ is trivial, and the case $n = 2$ is identical to the very definition of convexity. Suppose that we have already proved the claim up to $n-1 \geq 2$. Assume without loss of generality that $p_n < 1$. Then:

$$\sum_{i=1}^n p_i f(x_i) = p_n f(x_n) + \sum_{i=1}^{n-1} p_i f(x_i) = p_n f(x_n) + (1-p_n) \sum_{i=1}^{n-1} \frac{p_i}{1-p_n} f(x_i) \geq$$

$$p_n f(x_n) + (1-p_n) f\Big(\sum_{i=1}^{n-1} \frac{p_i}{1-p_n} x_i\Big) \geq f\Big(p_n x_n + (1-p_n) \sum_{i=1}^{n-1} \frac{p_i}{1-p_n} x_i\Big),$$

which is equal to $f(p_1 x_1 + \cdots + p_n x_n)$. That proves the claim. Note that the first inequality above follows from the induction hypothesis (the case $n-1$) and the second follows from the case $n = 2$. As to the strictness claim, if $x_1, \ldots, x_n$ are not all identical, then either $x_1, \ldots, x_{n-1}$ are not all identical and the first inequality is strict by induction hypothesis, or $x_1 = \cdots = x_{n-1} \neq x_n$ so that the last inequality is strict by definition. □

2.3 Bit Strings

For any $n \in \mathbb{N}$, let $\{0,1\}^n$ denote the n-fold Cartesian product of $\{0,1\}$. For $n = 0$, this is defined to be the set consisting of a special character $\perp$ only, the *empty string*. Define

$$\{0,1\}^* = \bigcup_{n \geq 0} \{0,1\}^n,$$

the set of (finite) *strings*. For strings $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_m)$ in $\{0,1\}^*$, the *concatenation* $x|y$ of x and y is defined to be the string

$$x|y = (x_1, \ldots, x_n, y_1, \ldots, y_m),$$

where, by definition, $x|\perp = x$ and $\perp|y = y$. Note that $(x|y)|z = x|(y|z)$ for all $x, y, z \in \{0,1\}^*$. When the meaning is clear from the context, we may also write xy instead of $x|y$, and for instance write 01001 instead of $(0,1,0,0,1)$.

For $x \in \{0,1\}^*$, the *length* $\ell(x)$ of x is defined to be the unique integer $n \in \mathbb{N}_0$ such that $x \in \{0,1\}^n$. It obviously holds that $\ell(\perp) = 0$ and

$$\ell(x|y) = \ell(x) + \ell(y)$$

for all $x, y \in \{0,1\}^*$.

A string $y \in \{0,1\}^*$ is called a *prefix* of $x \in \{0,1\}^*$ if there exists $z \in \{0,1\}^*$ such that $x = y|z$. Similarly, $y \in \{0,1\}^*$ is called a *suffix* of $x \in \{0,1\}^*$ if there exists $z \in \{0,1\}^*$ such that $x = z|y$. Obviously, the empty string $\perp$ is prefix and suffix of any $x \in \{0,1\}^*$.

3 Measures of Uncertainty

3.1 Shannon Entropy

Definition 2 *Let X be a random variable with image $\mathcal{X}$. The* (Shannon) entropy $H(X)$ *of X is defined as*

$$H(X) = \sum_{x\in\mathcal{X}} P_X(x)\cdot\log\frac{1}{P_X(x)} = -\sum_{x\in\mathcal{X}} P_X(x)\cdot\log P_X(x)\,,$$

with the convention that the corresponding argument in the summation is declared 0 for $x\in\mathcal{X}$ with $P_X(x)=0$ (which is justified by taking a limit).[1]

It is important to realize that the entropy of X is a function (solely) of the *distribution* P_X of X. However, it is customary to write $H(X)$ instead of the formally correct $H(P_X)$.

The entropy of X can also be expressed as the expectation of the random variable $\log\big(1/P_X(X)\big)$:

$$H(X) = E\Big[\log\frac{1}{P_X(X)}\Big]\,.$$

Proposition 3 *Let X be a random variable with image $\mathcal{X}$. Then*

$$0 \le H(X) \le \log(|\mathcal{X}|).$$

Equality on the left-hand side holds iff there exists $x\in\mathcal{X}$ with $P_X(x)=1$ (and thus $P_X(x')=0$ for all $x'\neq x$). Equality on the right-hand side holds iff $P_X(x)=1/|\mathcal{X}|$ for all $x\in\mathcal{X}$.

Proof. The function $f:\mathbb{R}_{>0}\longrightarrow\mathbb{R}$, $y\mapsto\log y$ is strictly concave on $\mathbb{R}_{>0}$. Thus, by Jensen's inequality:

$$H(X) = \sum_{x\in\mathcal{X}} P_X(x)\cdot\log\frac{1}{P_X(x)} \le \log\big(\textstyle\sum_{x\in\mathcal{X}} 1\big) = \log(|\mathcal{X}|).$$

Furthermore, since we may restrict the sum to all x with $P_X(x)>0$, equality holds if and only if $\log(1/P_X(x)) = \log(1/P_X(x'))$, and thus $P_X(x)=P_X(x')$, for all $x,x'\in\mathcal{X}$.

Finally, for the characterization of the lower bound, it is obvious that $H(X)=0$ if $P_X(x)=1$ for some x, and, on the other hand, if $H(X)=0$ then for any x with $P_X(x)>0$ it must be that $\log(1/P_X(x))=0$ and hence $P_X(x)=1$. □

For a *binary* random variable X, meaning that the image $\mathcal{X}$ of X consists only of two values $\mathcal{X}=\{x_0,x_1\}$, with probabilities $P_X(x_0)=p$ and $P_X(x_1)=1-p$, we can write $H(X)=h(p)$, where h denotes the *binary entropy function* defined as

$$h(q) = -\big(q\log(q)+(1-q)\log(1-q)\big)$$

for $0<q<1$ and $h(q)=0$ for $q=0$ or $q=1$.

3.2 Conditional Entropy

Let X be a random variable and $\mathcal{A}$ an event. Applying Definition 2 to the conditional probability distribution $P_{X|\mathcal{A}}$ naturally defines the entropy of X conditioned on the event $\mathcal{A}$ as

$$H(X|\mathcal{A}) = \sum_{x\in\mathcal{X}} P_{X|\mathcal{A}}(x)\cdot\log\frac{1}{P_{X|\mathcal{A}}(x)}.$$

[1] Shannon once said: *My greatest concern was what to call it. I thought of calling it* information*, but the word was overly used, so I decided to call it* uncertainty*. When I discussed it with John von Neumann, he had a better idea. Von Neumann told me: "You should call it* entropy*, for two reasons. In the first place, your uncertainty function has been used in statistical mechanics under that name, so it already has a name. In the second place, and more important, nobody knows what entropy really is, so in a debate you will always have the advantage."*

Definition 3 *Let X and Y be random variables, with respective images $\mathcal{X}$ and $\mathcal{Y}$. The* conditional entropy *$H(X|Y)$ of X given Y is defined as*

$$H(X|Y) = \sum_{y\in\mathcal{Y}} P_Y(y)\cdot H(X|Y{=}y)\,,$$

with the convention that the corresponding argument in the summation is 0 for $y \in \mathcal{Y}$ with $P_Y(y) = 0$.

Note that conditional entropy is not the entropy of a probability distribution but an expectation: the average uncertainty about X when given Y.

Proposition 4 *Let X and Y be random variables with respective images $\mathcal{X}$ and $\mathcal{Y}$. Then*

$$0 \le H(X|Y) \le H(X)$$

Equality on the left-hand side holds iff X is determined by Y, i.e., for all $y \in \mathcal{Y}$, there is an $x \in \mathcal{X}$ such that $P_{X|Y}(x|y) = 1$. Equality on the right-hand side holds iff X and Y are independent.

The upper bound expresses that (on average!) additional information, i.e. knowing Y, can only *decrease* the uncertainty.

Proof. The lower bound follows trivially from the definition and from Proposition 3, and so does the characterization of when $H(X|Y) = 0$. For the upper bound, note that

$$H(X|Y) = \sum_y P_Y(y) \sum_x P_{X|Y}(x|y) \log \frac{1}{P_{X|Y}(x|y)} = \sum_{x,y} P_{XY}(x,y) \log \frac{P_Y(y)}{P_{XY}(x,y)}$$

and

$$H(X) = \sum_x P_X(x) \log \frac{1}{P_X(x)} = \sum_{x,y} P_{XY}(x,y) \log \frac{1}{P_X(x)}$$

where in both expressions, we may restrict the sum to those pairs (x, y) with $P_{XY}(x, y) > 0$. Using Jensen's inequality, it follows that

$$\begin{aligned} H(X|Y) - H(X) &= \sum_{x,y} P_{XY}(x,y) \log \frac{P_X(x)P_Y(y)}{P_{XY}(x,y)} \\ &\le \log\Big(\sum_{x,y} P_X(x)P_Y(y)\Big) \le \log\Big(\big(\sum_x P_X(x)\big)\big(\sum_y P_Y(y)\big)\Big) = \log 1 = 0\,. \end{aligned}$$

Note that in the second inequality, we replaced the summation over all (x, y) with $P_{XY}(x, y) > 0$ by the summation over all $(x, y) \in \mathcal{X} \times \mathcal{Y}$.

For the first inequality, equality holds if and only if $P_{XY}(x, y) = P_X(x)P_Y(y)$ for all (x, y) with $P_{XY}(x, y) > 0$, and for the second inequality, equality holds if and only if $P_{XY}(x, y) = 0$ implies $P_X(x)P_Y(y) = 0$ for any $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. It follows that $H(X|Y) = H(X)$ if and only if $P_{XY}(x, y) = P_X(x)P_Y(y)$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. □

Proposition 5 (Chain Rule) *Let X and Y be random variables. Then*

$$H(XY) = H(X) + H(Y|X)\,.$$

In particular,

$$H(XY) \le H(X) + H(Y)\,.$$

The second property is called *subadditivity* of the entropy.

Proof. The inequality follows from $H(Y|X) \le H(Y)$. The chain rule itself is a simple matter of rewriting:

$$
\begin{aligned}
H(XY) &= -\sum_{x,y} P_{XY}(x,y) \log P_{XY}(x,y) \\
&= -\sum_{x,y} P_{XY}(x,y) \log\big(P_X(x) P_{Y|X}(y|x)\big) \\
&= -\sum_{x,y} P_{XY}(x,y) \log P_X(x) - \sum_{x,y} P_{XY}(x,y) \log P_{Y|X}(y|x) \\
&= -\sum_{x} P_X(x) \log P_X(x) - \sum_{x} P_X(x) \sum_{y} P_{Y|X}(y|x) \log P_{Y|X}(y|x) \\
&= H(X) + H(Y|X)\,.
\end{aligned}
$$

This was to be shown. □

Note that applying Definition 3 to the conditional distribution $P_{XY|\mathcal{A}}$ naturally defines $H(X|Y,\mathcal{A})$, the entropy of X given Y and conditioned on the event $\mathcal{A}$. Since the entropy is a function of the distribution of a random variable, the chain rule also holds when conditioning on an event $\mathcal{A}$. Furthermore, it holds that

$$H(X|YZ) = \sum_z P_Z(z) H(X|Y, Z{=}z)\,,$$

which is straightforward to verify. With this observation, it is easy to see that the chain rule generalizes as follows.

Corollary 1 *Let X, Y and Z be random variables. Then*

$$H(XY|Z) = H(X|Z) + H(Y|XZ)\,.$$

Inductively applying the (generalized) chain rule implies that for any sequence $X_1, \ldots, X_n$ of random variables:

$$H(X_1 \cdots X_n) = H(X_1) + H(X_2|X_1) + \cdots + H(X_n|X_{n-1} \cdots X_1)\,.$$

3.3 Mutual Information

Definition 4 *Let X and Y be random variables. The* mutual information $I(X;Y)$ *of X and Y is defined as*

$$I(X;Y) = H(X) - H(X|Y).$$

Thus, in a sense, mutual information reflects the reduction in uncertainty about X when given Y. Note that the mutual information is symmetric and non-negative:

$$I(X;Y) = H(X) - H(X|Y) = H(X) - H(XY) + H(Y) = H(Y) - H(Y|X) \geq 0,$$

with equality if and only if X and Y are independent. This follows immediately from the properties of the conditional entropy.

Applying Definition 4 to the conditional distribution $P_{XY|\mathcal{A}}$ naturally defines $I(X;Y|\mathcal{A})$, the mutual information of X and Y conditioned on the event $\mathcal{A}$.

Definition 5 *Let X, Y, Z be random variables. Then the* conditional mutual information of X and Y given Z *is defined as*

$$I(X;Y|Z) = \sum_z P_Z(z) I(X;Y|Z{=}z)\,,$$

with the convention that the corresponding argument in the summation is 0 for z with $P_Z(z) = 0$.

Obviously, $I(X;Y|Z)$ is symmetric in X and Y, and

$$I(X;Y|Z) \geq 0\,.$$

Furthermore, the previous bounds $H(X) \geq 0$, $H(X|Y) \geq 0$, and $I(X;Y) \geq 0$, can all be seen as special cases of $I(X;Y|Z) \geq 0$. These bounds, and any bound they imply, are called *Shannon inequalities*. We will later see that there also exist *non-Shannon inequalities*.

It is important to realize that $I(X;Y|Z)$ may be larger or smaller than (or equal to) $I(X;Y)$. The following is easy to verify (and is sometimes used as definition of $I(X;Y|Z)$).

Lemma 2 *Let X, Y, Z be random variables. Then*

$$I(X;Y|Z) = H(X|Z) - H(X|YZ)\,.$$

By this, and then applying the (generalized) chain rule, we obtain:

Corollary 2 *Let W, X, Y and Z be random variables. Then*

$$I(WX;Y|Z) = I(X;Y|Z) + I(W;Y|ZX)\,.$$

3.4 Entropy Diagrams

For two and three random variables, the relations between the different information-theoretic measures can be nicely represented by means of a Venn-diagram-like *entropy diagram*. The case of two random variables is illustrated in Figure 1 (left). From the diagram, one can for instance easily read off the relations $H(X|Y) \leq H(X)$, $I(X;Y) = H(X) + H(Y) - H(XY)$ etc. The case of three random variables is illustrated in Figure 1 (right).

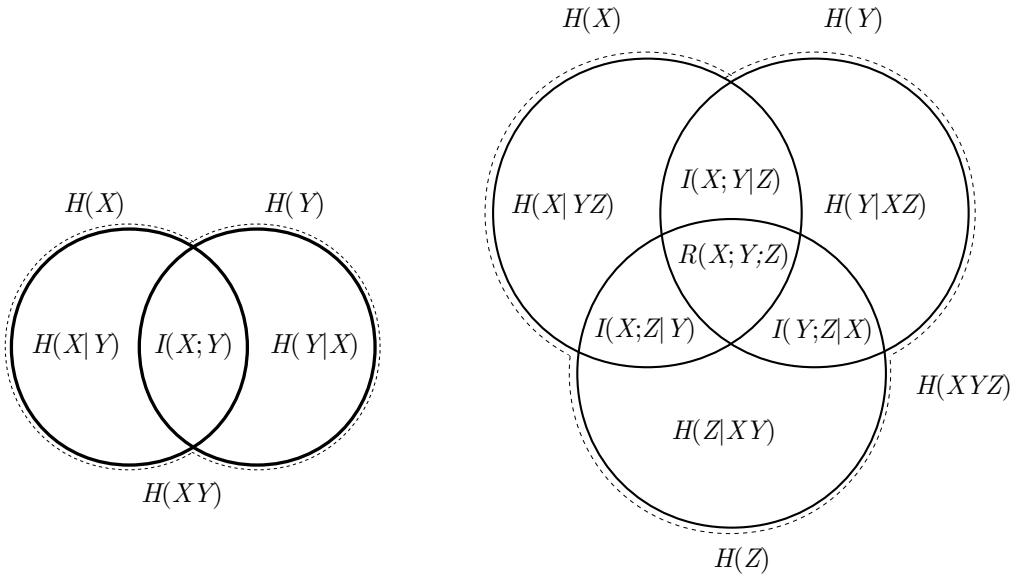


Figure 1: Entropy diagram for two (left) and three (right) random variables.

Also here, one can easily read off all the relations between the information-theoretic measures, like for instance $H(X|YZ) = H(X) - I(X;Z) - I(X;Y|Z)$, which is a relation that is otherwise maybe not immediately obvious. One subtlety with the entropy diagram for three random variables is that the "area in the middle", $R(X;Y;Z) = I(X;Y) - I(X;Y|Z)$, may be *negative*.

One may even consider an entropy diagram for four random variables, as illustrated in Figure 2, but here one has to be very cautious because various areas in the diagram may be negative.

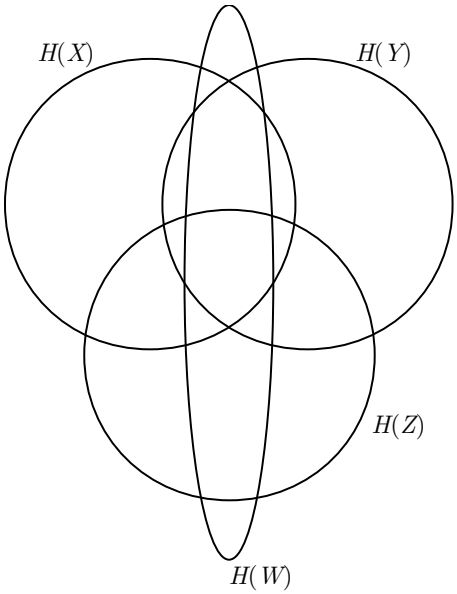


Figure 2: Entropy diagram for four random variables.

3.5 Non-Shannon Inequalities

Let n be a positive integer. For any list $X_1, \ldots, X_n$ of random variables, the values of all the joint entropies $H(X_1), \ldots, H(X_n), H(X_1X_2), \ldots, H(X_{n-1}X_n), \ldots H(X_1 \cdots X_n)$ fix a point p in the (2^n-1)-dimensional space $\mathbb{R}^{2^n-1}$, simply by assigning these values to the 2^n-1 coordinates of p. Let $\Gamma_n^* \subset \mathbb{R}^{2^n-1}$ be the set of all such points obtained by quantifying over all lists of random variables $X_1, \ldots, X_n$ (actually: their joint distributions) with arbitrary images. The question of interest is: What does Γ_n^* look like? Which points $p \in \mathbb{R}^{2^n-1}$ can be obtained from a list $X_1, \ldots, X_n$ of random variables in the above way?

Obviously, Γ_n^* is restricted by the *Shannon inequalities*, i.e., by the inequalities $I(U;V|W) \geq 0$, where U, V and W may consist of arbitrary (and possibly empty and/or overlapping) collections of the random variables $X_1, \ldots, X_n$.[2] For example, any point $p = (p_1, p_2, \ldots, p_{2^n-1}) \in \Gamma_n^*$ must satisfy $p_1 + p_2 - p_{n+1} \geq 0$, expressing that $I(X_1;X_2) = H(X_1) + H(X_2) - H(X_1X_2) \geq 0$. Hence, if Γ_n denotes the set of all points $p \in \mathbb{R}^{2^n-1}$ for which the Shannon inequalities hold, then $\Gamma_n^* \subseteq \Gamma_n$. But what about the converse? It can been shown that $\bar{\Gamma}_n^*$, the closure of Γ_n^*, is a *convex cone*, which means that it is fully characterized by inequalities of the same *linear* form

$$\sum_i \lambda_i H(X_i) + \sum_{i<j} \lambda_{i,j} H(X_iX_j) + \ldots + \lambda_{1,\ldots,n} H(X_1 \cdots X_n) \geq 0$$

as the Shannon inequalities. But are the Shannon inequalities *sufficient* to specify Γ_n^* or $\bar{\Gamma}_n^*$?

For $n = 2$ and 3, the answer is: yes. Indeed, it holds that $\Gamma_2^* = \bar{\Gamma}_2^* = \Gamma_2$ and $\bar{\Gamma}_3^* = \Gamma_3$. For $n > 3$, the answer had long been unknown (and was finally solved by Zhang and Yeung in 1998). Below, we show that for $n > 3$ there do exist *non-Shannon inequalities*, i.e., inequalities of the above linear form, that do not follow from the Shannon inequalities, yet need to be satisfied for any list $X_1, \ldots, X_n$ of random variables.

Proposition 6 *For any list of five random variables U, V, X, Y, Z:*

$$H(XY) + 2I(U;V) \leq H(X) + H(Y) + I(U;V|X) + I(U;V|Y) + 2I(U;V|Z) + I(UV;Z)\,.$$

By letting $X = Z$ we also get

[2] Note that $I(U;V|W)$ is determined by (a subset of) the values of $H(U), H(V), H(W), H(UV), H(UW), H(VW)$ and $H(UVW)$.

Corollary 3 *For any list of four random variables U, V, X, Y:*

$$H(XY) + 2I(U;V) \leq H(X) + H(Y) + 3I(U;V|X) + I(U;V|Y) + I(UV;X)\,.$$

Before we prove Proposition 6, we show that the inequality from Corollary 3, and thus also the inequality from Proposition 6, does not follow from the Shannon inequalities, and thus is non-Shannon. We do this by assigning values to all the quantities $H(U), H(V), H(X), H(Y), H(UV), \ldots, H(UVXY)$ in such a way that all Shannon inequalities are satisfied, but the inequality from Corollary 3 is not. A suitable assignment is as follows:

$$\begin{aligned}
&H(U) = H(V) = H(X) = H(Y) = 2\,,\\
&H(UV) = H(UX) = H(UY) = H(VX) = H(VY) = 3\,,\\
&H(XY) = 4\,,\text{ and}\\
&H(UVX) = H(UVY) = H(UXY) = H(VXY) = H(UVXY) = 4\,.
\end{aligned}$$

It is straightforward to verify that with this assignment, the inequality from Corollary 3 is violated (with a gap of size 1). Also, it is straightforward to verify that this assignment is consistent with the entropy diagrams in Figure 3. Since all the Shannon inequalities involving U, V, X, Y can be read out from these diagrams (noting the symmetries between U and Y and between X and Y), it follows that all Shannon inequalities are satisfied. We can thus conclude that the inequality from Corollary 3 is a non-Shannon inequality and that $\bar{\Gamma}_4^* \neq \Gamma_4$.[3]

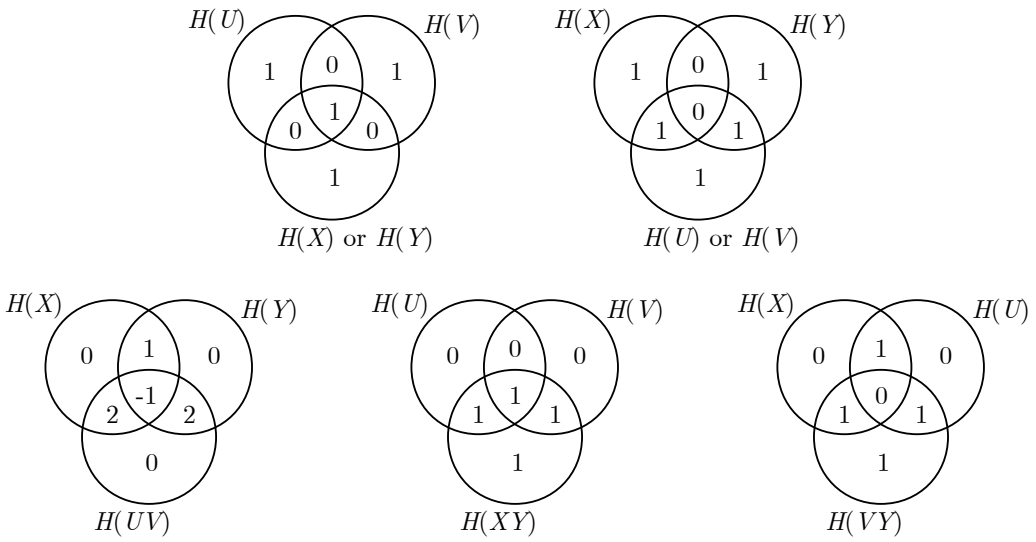


Figure 3: Entropy diagrams for the specific assignment.

Proof (of Proposition 6). First, we prove the inequality under the assumption that $I(XY;Z|UV) = 0$. Using the entropy diagram for 4 random random variables (as in Figure 2), and noting that the assumption in particular implies that $I(X;Z|UV) = 0$, one can easily show that

$$I(U;V) \leq I(U;V|X) + I(U;V|Z) + I(X;Z)\,.$$

Due to the symmetry in X and Y, it also holds that

$$I(U;V) \leq I(U;V|Y) + I(U;V|Z) + I(Y;Z)\,.$$

Finally, we can bound $H(XY)$ as

$$H(XY) = H(XY|Z) + I(XY;Z) \leq H(X|Z) + H(Y|Z) + I(UV;Z)$$

[3]The claim for Γ_4^* follows immediately; the claim for its closure follows from the smoothness of the entropy function.

where the inequality follows from the subadditivity of the (conditional) entropy, and from $I(XY;Z) \leq I(UV;Z)$, which follows from the assumption on $I(XY;Z|UV)$ as can easily be seen by means of the entropy diagram for the three random variables XY, Z and UV. Adding up the three inequalities results in the claimed inequality.

Now, the assumption on $I(XY;Z|UV)$ needs to be removed. The crucial observation is that in all the terms in the inequality, Z only appears in combination with U and V, but never in combination with X or Y. This allows us to re-define Z in such a way that the joint distribution of U, V and Z does not change, and thus all the terms in the inequality do not change their value, and at the same time $I(XY;Z|UV)$ vanishes, and thus the inequality holds by the above analysis. Formally, we extend the probability space given by U, V, X, Y and introduce a new random variable Z', defined by its conditional distribution

$$P_{Z'|UVXY}(z|u,v,x,y) = P_{Z|UV}(z|u,v)$$

for all z', u, v, x, y. By construction: $I(XY;Z'|UV) = 0$, and therefore the inequality holds with Z replaced by Z'. Furthermore, again by construction: $P_{Z'UV} = P_{ZUV}$, and therefore $I(U;V|Z') = I(U;V|Z)$ and $I(UV;Z') = I(UV;Z)$. It follows that the inequality holds also for the original Z. □

4 Perfectly Secure Encryption

A classical encryption scheme consists of finite non-empty sets $\mathcal{M}$, $\mathcal{K}$, and $\mathcal{C}$, and a function

$$e : \mathcal{M} \times \mathcal{K} \longrightarrow \mathcal{C}$$

such that for each $k \in \mathcal{K}$, the function $e(k,\cdot) : \mathcal{M} \longrightarrow \mathcal{C}$ is injective. For the purposes of the negative result below we introduce a more general definition.

Definition 6 *An encryption scheme for a random variable M, the message, consists of random variables K and C, the key and the ciphertext, respectively, such that M and K are independent and*

$$H(M|KC) = 0\,.$$

The condition on the conditional entropy repeats, in the language of information theory, that there is always unique decryption of the ciphertext. The definition also allows additional randomness in the computation of the ciphertext, besides the secret key. Moreover, it doesn't require that the secret key is chosen with uniform distribution.

For an encryption scheme in the classical sense, given by $\mathcal{M}, \mathcal{K}, \mathcal{C}$ and e, K and C are given as follows. K is the uniform distribution over $\mathcal{K}$, and C is specified by $C = e(K, M)$.

For an encryption scheme to be perfectly secure, we require that the ciphertext gives no information whatsoever about the plaintext: the uncertainty about the plaintext is as large as it is without knowing any ciphertext.

Definition 7 *An encryption scheme K, C for message M is* perfectly secure *if*

$$I(M;C) = 0\,.$$

An example of a perfectly secure encryption scheme is the one-time pad. In the one-time pad scheme, the sets $\mathcal{M}, \mathcal{K}, \mathcal{C}$ are all identified with the same finite additive group G, for instance $G = \{0,1\}^n$ with the bit-wise XOR as group operation. The encryption function e is then simply defined as

$$e(k,m) = k + m \in G\,.$$

The key k is chosen according to the uniform distribution over $\mathcal{K}$, independently from anything else.

Theorem 1 *The one-time pad scheme is perfectly secure.*

The main drawback of the one-time pad scheme is that the key is as large as the message that is to be encrypted. Shannon's pessimistic result states that this is inherent to any perfectly secure encryption scheme:

Theorem 2 *For any perfectly secure encryption scheme it holds that $H(K) \geq H(M)$.*

Both claims can be proven easily by means of the entropy diagram for three random variables.

5 Data Compression

A trivial counting argument shows that it is possible to encode the elements of a set $\mathcal{X}$ by bit strings of length n, where $n = \lceil \log(|\mathcal{X}|) \rceil$. Thus, to store or to transmit an element $x \in \mathcal{X}$, n bits of information always suffice. However, it should be clear that if not all $x \in \mathcal{X}$ are equally likely, in the sense that certain elements have a much higher probability than others, one should be able to exploit this and to achieve codes with shorter *average length*. The idea is of course to use encodings of varying lengths, assigning shorter codewords to the elements in $\mathcal{X}$ that have higher probabilities, and vice versa. The question we answer in this section is: how short can such a code be (on average over the choice of x)?

A necessary condition on an encoding function mapping elements of $\mathcal{X}$ to bit strings of finite length is that it is injective. However, if one transmits a sequence $x_1, \ldots, x_m \in \mathcal{X}$ (or stores them "sequentially") by sending one long concatenation $C(x_1)| \cdots |C(x_m)$ of the bit strings $C(x_i)$ encoding the x_i's, ambiguities may arise, namely in cases where it is possible to parse this long string in two consistent but different ways. Indeed, injectivity of the encoding function per se does not rule out that there exists a positive integer m' and elements $x'_1, \ldots, x'_{m'} \in \mathcal{X}$ such that

$$C(x_1)| \cdots |C(x_m) = C(x'_1)| \cdots |C(x'_{m'}).$$

An code with the guarantee that the encoding of any sequence of elements can be uniquely parsed is called *uniquely decodable*. Or, said in a different way, the encoding function C induces a map C^* in the obvious way from the set of all finite length sequences of elements from $\mathcal{X}$ into the set of all bit strings of finite length, and unique decodability means that C^* is injective as well. Of course, the problem of unique decodability of a list of codewords could be circumvented by introducing a special separation symbol. However, such a symbol might not be available, and maybe even more importantly, if an additional symbol *is* available, then one can do better by using a good uniquely decodable encoding of $\mathcal{X}$ into strings made up of bits and the additional symbol.

One convenient way to guarantee unique decodability is to require code to be *prefix-free*. This means that $C(x)$ is not a prefix of $C(x')$ for any distinct $x, x' \in \mathcal{X}$. With a prefix-free encoding, the elements $x_1, \ldots, x_m$ can be uniquely recovered from $C(x_1)| \cdots |C(x_m)$, simply by reading the encoding from left to right one bit at a time: by prefix-freeness it will remain unambiguous as reading continues when the current word terminates and the next begins. Thus, a prefix-free code is also appealing from an efficiency point of view, as it allows to decode "on the fly". For a general uniquely decodable code one may possibly have to inspect all bits in the entire string before being able to even recover the first word.

As argued, prefix-freeness is a nice feature, but it is also considerably more restrictive than mere unique decodability; thus, it is natural to ask: how much do we lose (in terms of the average codeword length) by requiring the encoding to be prefix-free rather than merely uniquely decodable? Surprisingly, the answer is: *nothing*. Indeed, we will show below that the length of an optimal prefix-free code and the length of an optimal uniquely decodable code coincide and are essentially given by the Shannon entropy.

5.1 Uniquely Decodable and Prefix-Free Codes

Let X be a random variable with image $\mathcal{X}$. We typically call X a *source*.

Definition 8 *A* source code*, or simply a* code*, for X is an injective function $C : \mathcal{X} \to \{0,1\}^*$. Such a code C is called* uniquely decodable *if the naturally induced function $C^* : \mathcal{X}^* \to \{0,1\}^*$ with $(x_1, \ldots, x_n) \mapsto C(x_1)| \ldots |C(x_n)$ is injective, where $\mathcal{X}^* = \bigcup_{n \in \mathbb{N}} \mathcal{X}^n \cup \{\perp\}$. Furthermore, C is called* prefix-free *if no $c \in \mathrm{im}(C)$ is prefix of a different $c' \in \mathrm{im}(C)$.*

We often refer to the set of codewords, $\mathcal{C} = \mathrm{im}(C)$, as code and leave the actual encoding function C implicit.

The following is easy to prove.

Lemma 3 *If a code $\mathcal{C}$ is prefix-free and $\mathcal{C} \neq \{\perp\}$ then $\mathcal{C}$ is uniquely decodable.*

Definition 9 *The* (average) length *of a code C for a source X is defined as*

$$\ell_C(X) = E[\ell(C(X))] = \sum_{x\in\mathcal{X}} P_X(x)\ell(C(x))\,.$$

Furthermore, we define the minimal code length *of a source X as*

$$\ell_{\min}^{\text{p.f.}}(X) = \min_{C\in\mathfrak{C}^{\text{p.f.}}} \ell_C(X) \qquad \textit{and} \qquad \ell_{\min}^{\text{u.d.}}(X) = \min_{C\in\mathfrak{C}^{\text{u.d.}}} \ell_C(X)$$

where $\mathfrak{C}^{\text{p.f.}}$ is the set of all prefix-free and $\mathfrak{C}^{\text{u.d.}}$ the set of all uniquely decodable codes $C : \mathcal{X} \to \{0,1\}^*$.

We will soon see that $\ell_{\min}^{\text{p.f.}}(X) = \ell_{\min}^{\text{u.d.}}(X)$, and thus we will just write $\ell_{\min}(X)$ from now on. A code C for which $\ell_C(X) = \ell_{\min}(X)$ is called *optimal* (for the source X).

5.2 Kraft's Inequality and Shannon's Theorem

The main theorem of this section shows that the minimal code length of a source X is essentially given by its entropy.

Theorem 3 (Shannon) *For any source X:*

$$H(X) \le \ell_{\min}(X) \le H(X) + 1\,.$$

The proof relies on the following theorem, known as Kraft's inequality.

Theorem 4 (Kraft's inequality) *There exists a prefix-free code $\mathcal{C} = \{c_1, \dots, c_m\}$ with codeword lenghts $\ell(c_1) = \ell_1, \dots, \ell(c_m) = \ell_m \in \mathbb{N}_0$ if and only if*

$$\sum_{i=1}^{m} \frac{1}{2^{\ell_i}} \le 1\,.$$

We will see that the forward implication holds also for uniquely decodable codes; the backward implication for uniquely decodable codes follows trivially from the original statement of Theorem 4 (unless $m = 1$ and $\ell_1 = 0$) due to Lemma 3. This version of Kraft's inequality is called *McMillan inequality*. It follows that for every uniquely decodable code there exists a prefix-free code with the very same codeword lengths, and thus in particular $\ell_{\min}^{\text{p.f.}}(X) = \ell_{\min}^{\text{u.d.}}(X)$.

We will now first prove Shannon's Theorem by using Kraft's inequality, and after that we will prove Kraft's inequality.

Proof (of Theorem 3). For the lower bound, let us fix the following notation. For any $x \in \mathcal{X}$, we write $\ell_x = \ell(C(x))$, and $\tilde{\mathcal{X}}$ denotes the set of all $x \in \mathcal{X}$ with $P_X(x) > 0$. We can thus write

$$\begin{aligned} H(X) - \ell_C(X) &= -\sum_{x\in\tilde{\mathcal{X}}} P_X(x)\log P_X(x) - \sum_{x\in\tilde{\mathcal{X}}} P_X(x)\ell_x \\ &= \sum_{x\in\tilde{\mathcal{X}}} P_X(x)\log\frac{1}{P_X(x)\cdot 2^{\ell_x}} \le \sum_{x\in\tilde{\mathcal{X}}} \log\frac{1}{2^{\ell_x}} \le \sum_{x\in\mathcal{X}} \log\frac{1}{2^{\ell_x}} \le \log(1) = 0\,, \end{aligned}$$

using Jensen's inequality and Kraft's inequality.

For the upper bound, let us define for any $x \in \mathcal{X}$

$$\ell_x = \left\lceil \log\frac{1}{P_X(x)} \right\rceil .$$

Note that

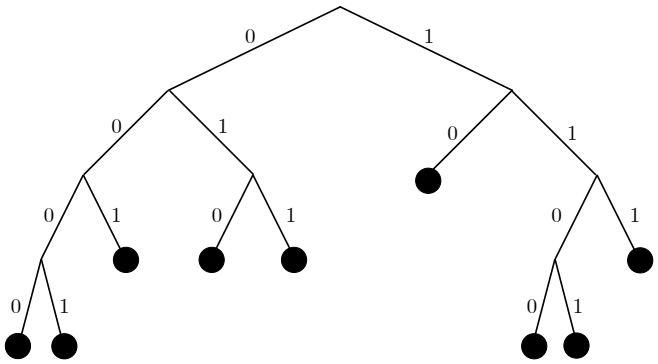
$$\sum_x \frac{1}{2^{\ell_x}} \le \sum_x P_X(x) = 1$$

and thus by Kraft's inequality, there exists a prefix-free code C such that $\ell(C(x)) = \ell_x$ for all $x \in \mathcal{X}$. This code satisfies

$$\ell_C(X) = \sum_x P_X(x)\ell_x \leq \sum_x P_X(x)\Big(\log \frac{1}{P_X(x)} + 1\Big) = -\sum_x P_X(x) \log P_X(x) + 1 = H(X) + 1\,.$$

This completes the proof. □

It remains to prove Kraft's inequality. The forward direction can easily be proven by noting that the codewords of a prefix-free code $\mathcal{C}$ correspond to (a subset of the) leaves in a binary tree, where every codeword $c \in \mathcal{C}$ can be found at depth $\ell(c)$ of the tree, and where the prefix-freeness ensures that no $c \in \mathcal{C}$ is a descendant of another $c' \in \mathcal{C}$ (see Figure 4 for an example).



$$\mathcal{C} = \{0000, 0001, 001, 010, 011, 10, 1100, 1101, 111\}$$

Figure 4: Example of a prefix-free code $\mathcal{C}$ as nodes in a binary tree.

For every $d \in \mathbb{N}_0$ and for every node in the tree (including but not restricted to the leaves) of depth d, one can now assign weight $1/2^d$ to that node. Noting that the weight of every node is the sum of the weights of its two children, one can now inductively "work its way to the root" and conclude that $\sum_c 1/2^{\ell(c)}$ is at most the weight of the root, which is 1.

In the proof given below, we prove a stronger statement, namely that the forward implication of Theorem 4 holds not only for prefix-free but even for uniquely decodable codes. Note that the implication holds trivially for the degenerated prefix-free code $\mathcal{C} = \{\bot\}$.

Proof (of Theorem 4). We start with the (strengthened) forward implication. Let $\mathcal{C}$ be a uniquely decodable code. We can write

$$S = \sum_{c \in \mathcal{C}} \frac{1}{2^{\ell(c)}} = \sum_{\ell=L_{\min}}^{L_{\max}} \frac{n_\ell}{2^\ell}$$

where $L_{\min} = \min_{c\in\mathcal{C}} \ell(c)$, $L_{\max} = \max_{c\in\mathcal{C}} \ell(c)$, and $n_\ell = |\{c \in \mathcal{C} \,|\, \ell(c) = \ell\}|$. Furthermore, for any $k \in \mathbb{N}$,

$$S^k = \sum_{c_1,\ldots,c_k \in \mathcal{C}} \frac{1}{2^{\ell(c_1)+\ldots+\ell(c_k)}} = \sum_{\ell=kL_{\min}}^{kL_{\max}} \frac{n_\ell^{(k)}}{2^\ell}$$

where $n_\ell^{(k)}$ is defined as $n_\ell^{(k)} = \big|\big\{(c_1,\ldots,c_k) \in \mathcal{C}^k \,\big|\, \sum_i \ell(c_i) = \ell(c_1|\ldots|c_k) = \ell\big\}\big|$. Note that

$$n_\ell^{(k)} = \sum_{x\in\{0,1\}^\ell} \big|\big\{(c_1,\ldots,c_k) \in \mathcal{C}^k \,\big|\, c_1|\ldots|c_k = x\big\}\big| \leq \sum_{x\in\{0,1\}^\ell} 1 = 2^\ell$$

where the inequality follows from the unique decodability of $\mathcal{C}$. Thus, we can conclude that

$$S^k \leq (L_{\max} - L_{\min}) \cdot k$$

for all $k \in \mathbb{N}$, from which follows that $S \leq 1$.

The backward implication, we prove by induction. The claim is trivial for $m = 1$ (and still easy to see for $m = 2$). For the induction step, let $m \geq 1$ and let $\ell_1 \leq \ldots \leq \ell_m \leq \ell_{m+1} \in \mathbb{N}$ be given. We distinguish between the following two cases.

Case 1: $\ell_m = \ell_{m+1}$. We apply the induction hypothesis to $\ell_1, \ldots, \ell_{m-1}, \ell'_m = \ell_m - 1$. Note that $2^{-\ell'_m} = 2^{-\ell_m} + 2^{-\ell_{m+1}}$ and hence the assumption (i.e. the bound on the sum) still holds. Let $\mathcal{C}' = \{c_1, \ldots, c_{m-1}, c'_m\}$ be the resulting prefix-free code. We now set $\mathcal{C} = \{c_1, \ldots, c_{m-1}, c_m, c_{m+1}\}$ where $c_m = c'_m|0$ and $c_{m+1} = c'_m|1$. It is straightforward to verify that $\mathcal{C}$ is still prefix-free.

Case 2: $\ell_m < \ell_{m+1}$. It is easy to see that $\ell_1, \ldots, \ell_m, \ell'_{m+1} = \ell_m$ still satisfies the bound on the sum. Indeed,

$$\sum_{i=1}^{m} \frac{1}{2^{\ell_i}} = \frac{\mu}{2^{\ell_m}}$$

for some $\mu \in \mathbb{N}$ which satisfies, by assumption, $\mu < 2^{\ell_m}$; therefore

$$\sum_{i=1}^{m} \frac{1}{2^{\ell_i}} + \frac{1}{2^{\ell'_{m+1}}} = \frac{\mu + 1}{2^{\ell_m}} \leq 1\,.$$

By the analysis of case 1, there exists a prefix-free code $\mathcal{C}' = \{c_1, \ldots, c_m, c'_{m+1}\}$ with codeword lengths $\ell_1, \ldots, \ell_m, \ell'_{m+1}$. We now set $\mathcal{C} = \{c_1, \ldots, c_m, c_{m+1}\}$ where $c_{m+1} = c'_{m+1}|0 \cdots 0$, with sufficiently many 0's padded. Again, it is straightforward to verify that $\mathcal{C}$ is still prefix-free. □

5.3 The Huffman Code

The constructive proofs for Theorem 3 and 4 show how to explicitly construct a code C for a source X such that $\ell_C(X)$ is off from the optimal $\ell_{\min}(X)$ by at most 1. However, C obtained this way is in general not optimal. In this section, we briefly describe a procedure for constructing an optimal prefix-free code C, i.e., one that achieves $\ell_C(X) = \ell_{\min}(X)$. The construction is due to David Huffman, and the resulting code C is called a *Huffman code*.

Let X be an arbitrary source with image $\mathcal{X}$. We may assume without loss of generality that $P_X(x) > 0$ for all $x \in \mathcal{X}$. Write $m = |\mathcal{X}|$. The construction is recursively. In case $m = 2$, simply assign the codewords 0 and 1 to the two elements in $\mathcal{X}$. In case $m > 2$, construct $\mathcal{C}$ as follows recursively. Let $x, x' \in \mathcal{X}$ have minimal probabilities, i.e., for all $y \in \mathcal{X}$ it holds that $P_X(y) \geq \max\{P_X(x), P_X(x')\}$. Reduce the number n of required code-words by "combining" x and x', resulting in a new symbol that occurs with probability $P_X(x) + P_X(x')$. This results in a new source with an image of size $m - 1$. Take now the Huffman code for this new source, and consider the code-word assigned to the new symbol, obtained by "combining" x and x'. Make two new words out of this one, by appending a 0, and by appending a 1. Assign these two new codewords to x and x'. This results in the Huffman code for X.

A somewhat more involved but much more informative description of the construction of Huffman codes is as follows. Take the initial probabilities as leaves in a tree that is to be constructed. In the first step, take the two nodes, i.e. leaves, with the smallest probabilities, and create an ancestor node in the tree for them. Assign to this node the sum of the two smallest probabilities. Now repeat this process, the list of nodes to which it is applied given by all the nodes with no ancestor (i.e., in the first iteration, the remaining leaves and the newly created node). Continue until a root has been created. Finally, the positions of the original leaves in the tree specify the corresponding codewords (similar to Figure 4).

Proving optimality of the Huffman code by induction is not too difficult but a bit tedious, and thus we do not do it here.

5.4 The Arithmetic Code

In this section, we present another explicit construction of a prefix-free code, known as *arithmetic* (or *Shannon-Fano-Elias*) *code*. In contrast to the Huffman code, the arithmetic code is in general

not optimal, but it has other advantages as we will discuss in the subsequent section, which usually makes it the better choice than the Huffman code for "real-life situations".

Let X be an arbitrary source with image $\mathcal{X}$. We may assume without loss of generality that $P_X(x) > 0$ for all $x \in \mathcal{X}$, and that $\mathcal{X} = \{1, \ldots, m\}$. Write $p_x = P_X(x)$ for any $x \in \mathcal{X}$. The encoding of an element $x \in \mathcal{X}$ is done as follows. The half-open interval $[0, 1) \subset \mathbb{R}$ is divided into m disjoint half-open intervals $I_1 = [a_1, b_1)$, $I_2 = [a_2, b_2)$, ..., $I_m = [a_m, b_m)$ such that I_j has size $b_j - a_j = p_j$ for any $j \in \mathcal{X}$, and $I_1 \cup \ldots \cup I_m = [0, 1)$. The encoding of an element $x \in \mathcal{X}$ is now defined to be the standard binary representation of some number in the interval I_x.

There are different possibilities on how to decide *which* number in I_x to choose. One possibility is to choose a number in I_x with the *smallest* binary representation, i.e., to let the encoding of x be the *shortest* string $c = (c_1, \ldots, c_\ell) \in \{0, 1\}^* \setminus \{\perp\}$ such that the number $\sum_{i=1}^{\ell} c_i \cdot 2^{-i}$ lies in I_x. Let us call the resulting code AC_0.

Proposition 7 *For any source X, the arithmetic code AC_0 has length*

$$\ell_{AC_0}(X) \leq H(X) + 1\,.$$

Proof. It is easy to see that any inverval $I = [a, b) \subseteq [0, 1)$ of size $p = b - a$ contains a number that has a binary representation $c = (c_1, \ldots, c_\ell)$ of length $\ell \leq \lceil \log(1/p) \rceil \leq -\log p + 1$. Indeed, one of the numbers $0 \cdot 2^{-\ell}, 1 \cdot 2^{-\ell}, 2 \cdot 2^{-\ell}, \ldots, (2^\ell - 1) \cdot 2^{-\ell}$ must lie in the interval I. It follows that

$$E\big[\ell\big(AC_0(X)\big)\big] = \sum_x P_X(x)\ell\big(AC_0(x)\big) \leq \sum_x P_X(x)(-\log p_x + 1) = H(X) + 1\,,$$

which was to be shown. □

The arithmetic code AC_0 described above is in general *not* prefix-free. However, we can easily make the code prefix-free by using a slightly different strategy for choosing c (actually: the number c represents) from the interval I_x. Instead of choosing a number in I_x with the *smallest* binary representation, c is chosen to be (the binary representation of) a number in *the lower half of* I_x with a binary representation of size equal to $\ell = \lceil \log(1/p) \rceil + 1$. It follows from the proof of Proposition 7 that such a c exists. Let us denote this code by $AC^{\text{p.f.}}$. It is easy to see that this modified code *is* prefix-free. Furthermore, the bound from Proposition 7 increases only by one, i.e., it holds that

$$\ell_{AC^{\text{p.f.}}}(X) \leq H(X) + 2\,,$$

as can easily be seen.

5.5 Encoding a Stream of Symbols

Consider the situation where the source is in fact a *stream* of symbols: $\boldsymbol{X} = X_1 \cdots X_n$, where all X_i's have the same image $\mathcal{X}$. Think of $\boldsymbol{X}$ as an English text, where $\mathcal{X}$ consists of the letters of the alphabet and some punctuation characters. Or, think of $\boldsymbol{X}$ as a picture file, where each X_i specifies the color of a different pixel.

Of course, in principle, one can use the Huffman code (or the arithmetic code) for $\boldsymbol{X}$ to encode the stream $X_1 \cdots X_n$. However, unless for small values of n, this is infeasible: en- and decoding requires an exponential amount of work (in n). Another, and more efficient, possibility is to encode the stream $X_1 \cdots X_n$ symbol-wise, i.e., to encode each X_i individually by means of a prefix-free code for X_i. If all X_i's have the same distribution, then the same code can be used for all the X_i's. The prefix-freeness ensures unique decodability. The drawback with this solution is that even when the optimal Huffman code is used to encode the X_i's, the expected codeword length may be as large as $H(X_1) + \cdots + H(X_n) + n$, compared to $H(X_1 \cdots X_n) + 1$, which can be achieved with an optimal code for $\boldsymbol{X}$. Thus, even if the X_i's are independent so that $H(X_1 \cdots X_n) = H(X_1) + \cdots + H(X_n)$, there is a potential overhead of up to $n - 1$ bits.

We show how the arithmetic code can be generalized to a code that encodes a stream $X_1 \cdots X_n$ of symbols in $\mathcal{X}$ in such a way that en- and decoding are (reasonably) efficient *and* the expected length

of the encoding has a small overhead. To encode $(x_1, \ldots, x_n) \in \mathcal{X}^n$, where we again assume that $\mathcal{X} = \{1, \ldots, m\}$, the following is done. First, as described in Section 5.4, the interval $[0, 1)$ is divided into m disjoint intervals $I_1, \ldots, I_m$, where I_j has size $P_{X_1}(j)$. But now, instead of (already) picking a number in the interval I_{x_1}, inductively for each $i \geq 1$ the interval $I_{x_1 \cdots x_i}$ is further divided up into m disjoint intervals $I_{x_1 \cdots x_i 1}, \ldots, I_{x_1 \cdots x_i m}$, where $I_{x_1 \cdots x_i \eta_{i+1}} = [a_{x_1 \cdots x_i \eta_{i+1}}, b_{x_1 \cdots x_i \eta_{i+1}})$ has *relative* size $(b_{x_1 \cdots x_i \eta_{i+1}} - a_{x_1 \cdots x_i \eta_{i+1}})/(b_{x_1 \cdots x_i} - a_{x_1 \cdots x_i}) = P_{X_{i+1}}(\eta_{i+1})$ for any $1 \leq \eta_{i+1} \leq m$; see Figure 5. In the end, the encoding c of $(x_1, \ldots, x_n)$ is set to be the standard binary representation of a number in the interval $I_{x_1 \cdots x_n}$ that has a minimal standard binary representation. Since this code is a generalization of the code AC_0 from Section 5.4, we also denote it by AC_0.

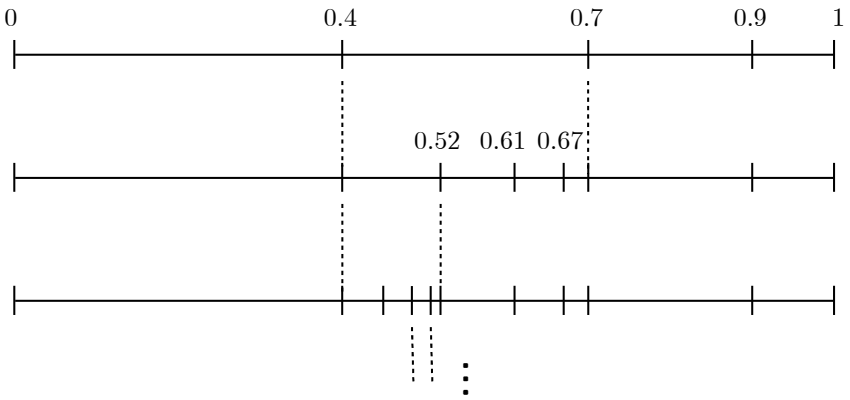


Figure 5: Arithmetic coding of $\boldsymbol{x} = (2, 1, 3, ...) \in \{1, 2, 3, 4\}^n$ with P_{X_i} given by $\frac{4}{10}$, $\frac{3}{10}$, $\frac{2}{10}$ and $\frac{1}{10}$.

If the decoder knows m, i.e., the number of symbols encoded, then decoding is (in principle) straightforward: inductively for $i \geq 1$ the decoder chooses x_i so that $I_{x_1 \cdots x_i}$ contains the number represented by c. There are some computational/algorithmic issues which we do not discuss here, but which can be solved. If the decoder does *not* know m, then he does not know when to stop. This stopping problem can be solved by different means, but for simplicity we just assume here that the decoder does know m.

The following is easy to prove.

Proposition 8 *For any stream $X_1 \cdots X_n$, the arithmetic code AC_0 has length*

$$\ell_{AC_0}(X_1 \cdots X_n) \leq H(X_1) + \ldots + H(X_n) + 1\,.$$

If the X_i's are independent, then this is close to optimal. However, if not then $\ell_{AC_0}(X_1 \cdots X_n)$ may still be far off from the optimal value, which is essentially given by $H(X_1 \cdots X_n)$. This is not too surprising since the arithmetic code as described above only considers the marginal distributions P_{X_1} up to P_{X_n} and completely ignores any possible relations between the X_i's. This is wasteful in many cases. For instance, in the English language the letter `e` is the most frequent letter, but if the previous letter was a `q`, then the letter `u` is much more likely to appear than an `e`.

Based on this observation, it makes sense to consider the following improved version of the arithmetic code. Instead of choosing $I_{x_1 \cdots x_i \eta_{i+1}}$ to be of relative size $P_{X_{i+1}}(\eta_{i+1})$, it is chosen to be of relative size $P_{X_{i+1}|X_i}(\eta_{i+1}|x_i)$. It follows that this improved version of the arithmetic code, which we call AC_1, satisfies

$$\ell_{AC_1}(X_1 \cdots X_n) \leq H(X_1) + H(X_2|X_1) + H(X_3|X_2) + \ldots + H(X_n|X_{n-1}) + 1\,.$$

Obviously, this can be further improved by taking not only the one previous but the *two* previous symbols into account, or in general the ℓ previous symbols, and choosing $I_{x_1 \cdots x_i \eta_{i+1}}$ to be of relative size $P_{X_{i+1}|X_i \cdots X_{i-\ell+1}}(\eta_{i+1}|x_i, \ldots, x_{i-\ell+1})$ etc. Note that the larger one wants to choose ℓ to get better compression, the better the source needs to be known by the encoder and the decoder, and the more computationally involved en- and decoding becomes.

5.6 Encoding an Unknown Source

So far we have assumed that encoder and decoder know $P_{X_1 \cdots X_n}$, or at least the $P_{X_{i+1}|X_i \cdots X_{i-\ell+1}}$'s for some fixed ℓ, or good approximations thereof. In many scenarios, this is not the case, or worse: these distributions are not even well defined. For instance, think of compressing a picture taken with your digital camera; what is the distribution for the color assignment of the pixels?

If no good *model* for the source exists or is known, then the following can be done. The approach is based on the assumption that the source $X_1, \ldots, X_n$ to be encoded forms a *stationary order-ℓ Markov chain* (for some ℓ), meaning that the conditional probability distribution function $P_{X_{i+1}|X_i \cdots X_{i-\ell+1}}$ is the same for every i; let us call this function D. Informally, this means that each symbol only depends on the ℓ previous symbols but not on the actual position within the stream. This is true in many natural scenarios, at least approximately. In this case, the encoder simply "computes" all values of the function $D = P_{X_{i+1}|X_j \cdots X_{i-\ell+1}}$ by counting the frequency of each symbol, the frequency of each pair of symbols, etc. within the actual stream $\boldsymbol{x} = (x_1, \ldots, x_n)$ that should be encoded. The computed conditional probability distribution function D is then used as model for the encoding of $\boldsymbol{x}$, where the encoding is done for instance by means of arithmetic encoding. Furthermore, the frequencies are encoded using a prefix-free encoding and prepended to the actual encoding of $\boldsymbol{x}$, so that also the decoder can "compute" D. Encoding the frequencies of the $(\ell+1)$-tuples of symbols requires approximately $|\mathcal{X}|^{\ell+1} \log(n)$ bits; thus, when $|\mathcal{X}|^{\ell+1}$ is small compared to n, this is an acceptable overhead.

Another approach is to start with a "weak" model (like uniform distribution for the X_i's), but let the encoder and decoder adaptively update the model, based on the symbols and frequencies observed so far. We do not discuss this in any more detail.

6 Guessing from Partial Information

Let X, Y be random variables. Suppose we are given Y. How much does that help us to guess the outcome of X. The following version of *Fano's inequality* gives an answer. Let $guess : \mathcal{Y} \to \mathcal{X}$ denote the "guessing function", and let the random variable $\hat{X} = guess(Y)$ capture the guess made at X given Y. Note that the optimal guessing strategy is to bet always on an element x of largest probability given y; however, this will not be crucial below. Finally, let p_e denote the (average) *error probability*

$$p_e = P[X \neq \hat{X}] \,.$$

Theorem 5 (Fano's inequality) *Let X, Y be random variables. Then for any function guess*

$$h(p_e) + p_e \log(|\mathcal{X}| - 1) \geq H(X|Y) \,,$$

and in particular if $|\mathcal{X}| = 2$ then

$$h(p_e) \geq H(X|Y) \,.$$

Recall that h denotes the binary entropy function as defined in Section 3.1. Also, note that there is equality if X represents the uniform distribution, and Y is absent (say by taking it equal to a constant random variable).

Proof. Define E as the binary random variable that returns 0 if $X = \hat{X}$, and 1 if $X \neq \hat{X}$. Note that $P_E(1) = p_e$ and thus $H(E) = h(p_e)$. By using the chain rule

$$H(EX|Y) = H(X|Y) + H(E|XY) \,,$$

but also

$$H(EX|Y) = H(E|Y) + H(X|EY) \,.$$

Clearly,

$$H(E|XY) = 0 \,,$$

since given X and Y, it is uniquely determined if $X = \hat{X}$ or not (for any fixed function *guess*). Also,

$$H(E|Y) \leq H(E),$$

since conditioning does not increase entropy. Finally

$$H(X|EY) = P_E(0) \cdot H(X|Y, E{=}0) + P_E(1) \cdot H(X|Y, E{=}1) \leq p_e \log(|\mathcal{X}| - 1)\,,$$

where the inequality follows from $H(X|Y, E{=}0) = 0$ and $H(X|Y, E{=}1) \leq \log(|\mathcal{X}| - 1)$. The former holds because if $E = 0$, i.e., $X = \hat{X}$, then X is uniquely determined by Y, and the latter holds because if $E = 1$, i.e., $X \neq \hat{X}$, then one value for X can be excluded and thus X can be one of at most $|\mathcal{X}| - 1$ values. Putting things together yields

$$H(E) + p_e \log(|\mathcal{X}| - 1) \geq H(X|Y),$$

which proves the claim. □

In the case where X is a bit string $X = (X_1, \ldots, X_k)$, we may also consider the *bit-error probability*

$$\bar{p}_e = \frac{1}{k} \sum_{i=1}^{k} P[X_i \neq \hat{X}_i]\,.$$

Corollary 4 *For X distributed over the k-bit strings: $h(\bar{p}_e) \geq \frac{1}{k} H(X|Y)$.*

Proof. This follows immediately from Theorem 5 as follows. Writing $p_{e,i} = P[X_i \neq \hat{X}_i]$, we can write

$$h(\bar{p}_e) = h\big(\tfrac{1}{k} \textstyle\sum_i p_{e,i}\big) \geq \frac{1}{k} \sum_i h(p_{e,i}) \geq \frac{1}{k} \sum_i H(X_i|Y) \geq \frac{1}{k} H(X|Y)$$

where the first inequality is due to Jensen's inequality, the second by Fano's inequality (Theorem 5), and the last one by the chain rule, noting that conditioning on more only reduces the entropy. □

7 Shannon's Channel Coding Theorem

A *channel* with (finite) respective input and output sets $\mathcal{X}$ and $\mathcal{Y}$ is given by a *conditional probability distribution* $P_{Y|X}$, which uniquely determines the distribution of the output of the channel when $x \in \mathcal{X}$ is input into the channel as $P_{Y|X}(\cdot|x)$.

Definition 10 *The* capacity *C of a channel given by $P_{Y|X}$ is defined as*

$$C = \max_{P_X} I(X;Y)\,.$$

We will show below that the capacity of a channel measures exactly how many bits of information on average can be reliably communicated per channel use.

Two simple examples are the *binary symmetric channel* (Figure 6, left) and the *binary erasure channel* (Figure 6, right). It is not too hard to see that the capacity of the binary symmetric channel with error probability ε is $C = 1 - h(\varepsilon)$, and the capacity of the binary erasure channel with erasure probability ε is $C = 1 - \varepsilon$.

Figure 6: The binary symmetric channel (left), and the binary erasure channel (right).

A k-bit *source* is given by k binary random variables $U_1, \ldots, U_k$. An *encoding scheme* for a k-bit source and a channel given by $P_{Y|X}$ consists of an encoding function $enc : \{0,1\}^k \to \mathcal{X}^n$ and a decoding function $dec : \mathcal{Y}^n \to \{0,1\}^k$. The random variables $X_1, \ldots, X_n$, $Y_1, \ldots, Y_n$, and $\hat{U}_1, \ldots, \hat{U}_k$ are then determined as pictured in Figure 7.

$$\boxed{\text{Source}} \to U_1,\ldots,U_k \to \boxed{\text{encode}} \to X_1,\ldots,X_n \to \boxed{\text{channel}} \to Y_1,\ldots,Y_n \to \boxed{\text{decode}} \to \hat{U}_1,\ldots,\hat{U}_k$$

Figure 7: Communication over a noisy channel.

For the negative result below (Theorem 7) we additionally allow *feedback*, meaning that X_{i+1} is computed as a function not only of $U_1, \ldots, U_k$, but actually as a function of the complete "history" $U_1, \ldots, U_k$, $X_1, \ldots, X_i$ and $Y_1, \ldots, Y_i$; thus, we assume that for each transmission the encoder learns what the receiver has received and may adapt his encoding strategy accordingly. This makes the negative result only stronger, and shows that feedback does not help to transmit more information (although it may make it easier to transmit that amount of information).

Definition 11 *The* rate *of an encoding scheme is defined as*

$$R = \frac{k}{n}\,,$$

i.e., as the number of bits communicated per channel use.

The error probability *and the* bit-error probability *of an encoding scheme, for a specific source, are respectively defined as*

$$p_e = P[\exists i : U_i \neq \hat{U}_i] \qquad \text{and} \qquad \bar{p}_e = \frac{1}{k}\sum_i P[U_i \neq \hat{U}_i]\,.$$

Note that obviously $p_e \geq \bar{p}_e$.

Theorem 6 (Channel-Coding Theorem - Part I) *For any channel with capacity C, and for any encoding scheme with rate $R > C$, the bit-error probability $\bar{p}_e$ is bounded as*

$$h(\bar{p}_e) \geq 1 - \frac{C}{R}\,.$$

Proof. In the following, superscripts mean that we consider the coordinates from 1 up to (and including) the superscript, e.g. $X^m = (X_1, \ldots, X_m)$. By using the chain rule, we can write

$$\begin{aligned} H(X^nY^nU^k) &= H(X^{n-1}Y^{n-1}U^k) + \overbrace{H(X_n|X^{n-1}Y^{n-1}U^k)}^{=0} + \overbrace{H(Y_n|X^nY^{n-1}U^k)}^{=H(Y_n|X_n)} \\ &= H(X^{n-1}Y^{n-1}U^k) + H(Y_n|X_n) \\ &= \cdots \\ &= H(U^k) + \sum_{i=1}^{n} H(Y_i|X_i)\,. \end{aligned}$$

But also, using again the chain rule,

$$H(X^nY^nU^k) = H(Y^nU^k) + H(X^n|Y^nU^k) = H(Y^nU^k)\,,$$

so that

$$H(Y^nU^k) = H(U^k) + \sum_{i=1}^{n} H(Y_i|X_i)\,.$$

Now we can bound the information Y^n gives about U^k as follows.

$$I(U^k;Y^n) = H(U^k) + H(Y^n) - H(Y^nU^k) = H(Y^n) - \sum_{i=1}^{n} H(Y_i|X_i)$$
$$\leq \sum_{i=1}^{n} \big(H(Y_i) - H(Y_i|X_i)\big) = \sum_{i=1}^{n} I(X_i;Y_i) \leq n \cdot C = k \cdot \frac{C}{R}$$

Therefore, for independent and uniformly distributed U_i's:

$$H(U^k|Y^n) = H(U^k) - I(Y^n;U^k) \geq k - k \cdot \frac{C}{R} = k\Big(1 - \frac{C}{R}\Big).$$

This already shows that from the point of view of the decoder, there is an inherent lower bound on the entropy of $U_1, \ldots, U_k$ per bit when $R > C$. The claim on the decoding error probability follows from Corollary 4. □

Theorem 7 (Channel-Coding Theorem - Part II) *For any channel with capacity C, and for any $R < C$ and $p_e > 0$, there exists an encoding scheme with rate at least R and error probability at most p_e for any distribution of the source.*

We only prove the special case of a binary symmetric channel with error probability ε, and thus capacity $C = 1 - h(\varepsilon)$. The proof for the general case can be found in Cover and Thomas [2].

Proof. We start by proving the claim for a *uniform* source.

For the construction of the encoding scheme, we are only concerned with finding a "good" set $\mathcal{C} \subset \{0,1\}^n$ of codewords; the encoding and decoding functions are then naturally given as follows. *enc* may be an arbitrary bijective map $enc : \{0,1\}^k \to \mathcal{C}$, and *dec* is such that for any $y \in \{0,1\}^n$, $dec(y)$ equals the string $u \in \{0,1\}^k$ such that $enc(u)$ is the closest codeword to y (with some fixed choice if there is an ambiguity).

We set $k = \lceil Rn \rceil$, where n will be chosen large enough. This guarantees the rate of the encoding scheme to be at least R. We construct $\mathcal{C} \subset \{0,1\}^n$ by choosing each codeword at random and independently from $\{0,1\}^n$.[4] We show that the error probability, where the probability is also over the random choice of $\mathcal{C}$, of the resulting encoding scheme is upper bounded by p_e (for a uniform source). This then implies that there exists a specific encoding scheme with error probability upper bounded by p_e (for a uniform source).

We let $c = c_1 \in \{0,1\}^n$ denote the codeword sent over the channel, $c' \in \{0,1\}^n$ the string received, and $c_2, \ldots, c_{2^k} \in \{0,1\}^n$ the remaining codewords. Recall that $c_2, \ldots, c_{2^k} \in \{0,1\}^n$ are random and independent of c and c' by construction. In the following, one should think of c as being fixed, the bounds on the probabilities then in particular also hold for a random c. Note that c' is obtained from c by flipping every bit independently with probability ε. Therefore, the expected Hamming distance $d_H(c,c')$ between c and c' is εn, and by the law of large numbers, $d_H(c,c')$ is close to εn except with arbitrary small probability for n large enough. In particular, for any $\alpha > 0$,

$$d_H(c,c') \leq (\varepsilon + \alpha)n$$

except with arbitrary small probability, for large enough n. Specifically, Hoeffding's inequality shows that the bound holds except with probability $e^{-2\alpha^2 n}$. On the other hand, from Lemma 4 below which bounds the number of strings contained in a ball of given radius, it follows that for any fixed $i \in \{2, \ldots, n\}$, the probability that

$$d_H(c_i,c') \leq (\varepsilon + \alpha)n$$

is upper bounded by $2^{h(\varepsilon+\alpha)n-n}$, and thus the probability that it holds for some non-specified i is upper bounded by

$$(2^k - 1) \cdot 2^{h(\varepsilon+\alpha)n-n} \leq 2^{k+h(\varepsilon+\alpha)n-n} \leq 2^{Rn+1+h(\varepsilon+\alpha)n-n} = 2 \cdot 2^{-(1-h(\varepsilon+\alpha)-R)n}$$

[4] Thus, we allow codewords to "collide", and therefore, formally, $\mathcal{C}$ should actually be viewed as a *multi*set.

which is exponentially small in n if $R < 1 - h(\varepsilon + \alpha)$, which can be achieved by a suitably small (but positive) choice for α if $R < 1 - h(\varepsilon) = C$. Therefore, if $R < C$ then the probability of incorrect decoding can be made as small as p_e by choosing n large enough.

The above probability holds for a randomly chosen set $\mathcal{C} = \{c_1, c_2, \ldots, c_{2^k}\}$ of codewords where the first codeword is transmitted, and thus also for a randomly chosen set of codewords where a *random* codeword is transmitted. It follows that there exists a specific set of codewords with error probability at most p_e when a random codeword is transmitted; we call such a codeword set to be *good on average*. It remains to argue that there exists a specific set of codewords that is *good in the worst-case*, meaning with error probability at most p_e for *any* codeword transmitted.

For given R and p_e, let $\mathcal{C}'$ be a set of codewords that is good on average, but constructed for parameters R' and p'_e such that $R < R' < C$ and $p'_e = p_e/2$. Since $R' > R$, for n large enough, $|\mathcal{C}'| = 2^{\lceil R'n \rceil} \geq 2 \cdot 2^{\lceil Rn \rceil}$. Furthermore, the number of codewords in $\mathcal{C}'$ for which the error probability is larger than $2p'_e$ is at most $|\mathcal{C}'|/2$. This means there are at least $2^{\lceil Rn \rceil}$ codewords in $\mathcal{C}'$ which have error probability at most $2p'_e = p_e$. Thus, these codewords form a code with rate at least R and error probability at most p_e for any code word, and thus for any distribution on the source. □

It remains to prove the following claim.

Lemma 4 *For any $0 \leq \delta \leq \frac{1}{2}$, any positive integer n and any n-bit string $w \in \{0,1\}^n$, the size of the set $B_{\delta n}(w) = \{v \in \{0,1\}^n \mid d_H(v,w) \leq \delta n\}$ is upper bounded by*

$$|B_{\delta n}(w)| \leq 2^{h(\delta)n}\,.$$

Proof. For the proof, note first that $|B_{\delta n}(w)|$, which does actually not depend on w and thus we may simply write $|B_{\delta n}|$, is given by

$$|B_{\delta n}| = \sum_{i=0}^{\lfloor \delta n \rfloor} \binom{n}{i}\,.$$

For simplicity, but actually without loss of generality, we may assume that δn is an integer. We can write

$$\begin{aligned} 1 = (\delta + (1-\delta))^n &= \sum_{i=0}^{n} \binom{n}{i} \delta^i (1-\delta)^{n-i} \geq \sum_{i=0}^{\delta n} \binom{n}{i} \delta^i (1-\delta)^{n-i} \\ &= (1-\delta)^n \sum_{i=0}^{\delta n} \binom{n}{i} \Big(\underbrace{\frac{\delta}{1-\delta}}_{\leq 1} \Big)^i \geq (1-\delta)^n \sum_{i=0}^{\delta n} \binom{n}{i} \Big(\frac{\delta}{1-\delta} \Big)^{\delta n} = \delta^{\delta n} (1-\delta)^{(1-\delta)n} \cdot |B_{\delta n}|\,. \end{aligned}$$

Taking logarithms finishes the proof. □

8 Almost Perfect Security

We have seen that information theory provides an adequate way to reason formally about perfectly secure encryption. The central concept is that of *no information* (at all!).

A random variable Y gives no information on a random variable X if X and Y are independent, or equivalently, if their mutual information $I(X;Y)$ is equal to 0. On the other hand, Y gives full information about X if the conditional entropy $H(X|Y)$ of X given Y is equal to 0. Also, $H(X) = \log |\mathcal{X}|$ in case of the uniform distribution on $\mathcal{X}$. Thus, if $I(X;Y) = 0$ and $H(X) = \log |\mathcal{X}|$, then Y gives no information on X, and X has the uniform distribution. If Y captures the information held by an adversary, and if X is a secret key, this would mean that from the point of view of the adversary, the secret key is maximally unpredictable.

What happens if in the discussion above the quantities are bounded away from their "optimal value" by a small ε? The answers to this question involves the concepts of *practically no information* and *almost random*. These concepts lead us to more realistic notions of security, which we develop below.

8.1 Statistical Distance and Indistinguishability

We first define what it means that two probability distributions are close, and then discuss a simple but crucial consequence.

Definition 12 *Let X and Y be random variables with the same image $\mathcal{V}$. The* statistical distance *$\Delta[X;Y]$ between X and Y (actually: between P_X and P_Y) is defined as*

$$\Delta[X;Y] = \frac{1}{2}\sum_{v\in\mathcal{V}} \big|P_X(v) - P_Y(v)\big| \,.$$

Note that this distance is a function of P_X and P_Y only; furthermore, the two distribution may also be obtained from different probability spaces. The statistical distance is a distance measure in the usual mathematical meaning, as becomes clear from the following lemma, which is straightforward to prove.

Lemma 5 *Let X, Y, Z be random variables with the same image $\mathcal{V}$. Then*

$$0 \le \Delta[X;Y] \le 1, \quad \Delta[X;X] = 0, \quad \Delta[X;Y] = \Delta[Y;X], \quad \textit{and} \quad \Delta[X;Z] \le \Delta[X,Y] + \Delta[Y,Z] \,.$$

We will also make use of the following property.

Lemma 6 *Let X and X' be random variables with image $\mathcal{X}$, and Y with image $\mathcal{Y}$. Then*

$$\Delta[XY;X'Y] = \sum_{y\in\mathcal{Y}} P_Y(y)\,\Delta[X;X'\,|Y{=}y] \,,$$

where $\Delta[X;X'\,|Y{=}y]$ is naturally defined as $\Delta[X;X'\,|Y{=}y] = \frac{1}{2}\sum_x \big|P_{X|Y}(x|y) - P_{X'|Y}(x|y)\big|$.

We want to argue that the statistical difference is "the right" distance measure for random variables (respectively probability distributions). One argument is that when we are given a "sample" $v \in \mathcal{V}$, chosen according to either the distribution P_X or the distribution P_Y, then the advantage of correctly distinguishing the two cases is at most $\Delta[X,Y]$.[5] This follows from the following fact.

Proposition 9 *Let X, Y be random variables with the same image $\mathcal{V}$. Then for any $\mathcal{W} \subset \mathcal{V}$ it holds that*

$$\Delta[X;Y] \ge |P_X(\mathcal{W}) - P_Y(\mathcal{W})|,$$

with equality iff $\mathcal{W}$ is the set $\{v \in \mathcal{V} : P_X(v) > P_Y(v)\}$ or its complement.

A short proof of this proposition can for instance be found in Shoup's book [3].

Another argument for why the statistical difference is "the right" distance measure, is as follows. Assume that X describes the behavior of the *real world* and Y the behavior of a hypothetical *ideal world*, which is defined to behave perfectly as desired. The following proposition now guarantees that we can think of these two worlds as co-existing in such a way that the real world looks exactly as the ideal world, except with probability $\Delta[X;Y]$. In particular, whatever happens in the real world also happens in the ideal world, except with probability $\Delta[X;Y]$, and vice versa. For instance, if the ideal world is such that some "bad" event never occurs, then that event occurs in the real world with probability at most $\Delta[X;Y]$. Thus, if $\Delta[X;Y]$ is "small" then we can conclude that the real world behaves essentially like the ideal world.

Proposition 10 *Let X and Y be random variables with the same image $\mathcal{V}$ and with respective distributions P_X and P_Y (possibly with respect to different probability spaces). Then there exists a joint probability distribution $Q_{XY}(x,y)$ on $\mathcal{V}\times\mathcal{V}$ such that the marginal distributions $Q_X(x) = \sum_{y\in\mathcal{V}} Q_{XY}(x,y)$ and $Q_Y(y) = \sum_{x\in\mathcal{V}} Q_{XY}(x,y)$ satisfy*

$$Q_X = P_X \qquad \textit{and} \qquad Q_Y = P_Y \,,$$

and such that

$$Q[X{\neq}Y] = \sum_{\substack{x,y\in\mathcal{V}\\ x\neq y}} Q_{XY}(x,y) \le \Delta[X;Y] \,.$$

[5] Formally, a distinguishing strategy is given by a subset $\mathcal{W} \subseteq \mathcal{V}$; $v \in \mathcal{W}$ is then interpreted as that P_X was used, and $v \notin \mathcal{W}$ is interpreted as that P_Y was used. The *advantage* for a strategy $\mathcal{W}$ is then defined as $|P_X(\mathcal{W}) - P_Y(\mathcal{W})|$.

Proof. We actually define a probability distribution $Q_{XYE}(x,y,e)$ on $\mathcal{V}\times\mathcal{V}\times\{0,1\}$, by introducing an extra binary random variable E. The claimed probability distribution will then be the marginal distribution $Q_{XY}(x,y) = Q_{XYE}(x,y,0) + Q_{XYE}(x,y,1)$. Set

$$Q_{XYE}(x,y,1) = \begin{cases} 0 & \text{if } x \neq y \\ \min\{P_X(x), P_Y(x)\} & \text{else} \end{cases}$$

and

$$Q_{XYE}(x,y,0) = \frac{\max\{P_X(x)-P_Y(x),0\}\cdot\max\{P_Y(y)-P_X(y),0\}}{\Delta[X;Y]}\,.$$

In the analysis below, we will use a couple of times the fact that $\Delta[X;Y]$ can also be written as $\Delta[X;Y] = \sum_v \max\{P_X(v)-P_Y(v),0\}$, which follows easily from the definition of $\Delta[X;Y]$.

Consider an arbitrary $x\in\mathcal{V}$. If $P_X(x)\leq P_Y(x)$ then $Q_{XE}(x,1) = P_X(x)$ and $Q_{XE}(x,0) = 0$, and thus $Q_X(x) = P_X(x)$. On the other hand, if $P_X(x)\geq P_Y(x)$ then $Q_{XE}(x,1) = P_Y(x)$ and

$$Q_{XE}(x,0) = (P_X(x)-P_Y(x))\frac{\sum_y \max\{P_Y(y)-P_X(y),0\}}{\Delta[X;Y]} = P_X(x)-P_Y(x)\,,$$

and thus, also here: $Q_X(x) = P_X(x)$. The corresponding can be shown for Q_Y, i.e., $Q_Y(y) = P_Y(y)$ for all $y\in\mathcal{V}$. Finally, it follows from the definition of E that if $X\neq Y$ then $E=0$, and therefore $Q[X\neq Y]\leq Q_E(0) = \sum_{x,y} Q_{XYE}(x,y,0) = \Delta[X;Y]\cdot\Delta[X;Y]/\Delta[X;Y] = \Delta[X;Y]$. □

8.2 Almost Uniform Distributions

We know that if a random variable X with image $\mathcal{X}$ satisfies $H(X) = \log|\mathcal{X}|$, then X is necessarily uniformly distributed (over $\mathcal{X}$). Therefore, one expects that if $H(X)$ is "almost" $\log|\mathcal{X}|$, then X must be "close" to uniformly distributed. The following result shows that this is indeed the case, where closeness is measured by means of the statistical distance.

Proposition 11 *Let X be a random variable with image $\mathcal{X}$, and let $H(X)\geq\log|\mathcal{X}|-\delta$. Then*

$$\Delta[X;U]\leq 2\ln 2\cdot\sqrt{\delta}$$

where U is uniformly distributed over $\mathcal{X}$.

The proof, which is based on the so-called Kullback-Leibler distance, can be found in Cover and Thomas [2]. [6]

The corresponding also holds in case of two random variables X and Y: if $H(X|Y)$ is "almost" $\log|\mathcal{X}|$, then X is "close" to uniformly distributed and independent of Y.

Corollary 5 *Let X and Y be a random variables with respective images $\mathcal{X}$ and $\mathcal{Y}$, and let $H(X|Y)\geq\log|\mathcal{X}|-\delta$. Then*

$$\Delta[XY;UY]\leq 2\ln 2\cdot\sqrt{\delta}$$

where U is uniformly distributed over $\mathcal{X}$ and independent of Y.

The claim follows from Lemma 6, and applying Proposition 11 and using Jensen's inequality.

[6] Note that Cover and Thomas speak of *variational distance* instead of statistical distance. That coincides with what some others call the L_1-distance, and it is exactly a multiplicative factor of 2 larger than statistical distance.

9 Randomness Extraction and Privacy Amplification

9.1 More Measures of Uncertainty

Definition 13 *Let X be a random variable with image $\mathcal{X}$. The* collision probability $\mathrm{Col}(X)$ *of X is defined as*

$$\mathrm{Col}(X) = \sum_{x\in\mathcal{X}} P_X(x)^2\,.$$

and the Rényi-entropy *(of order 2) or* collision-entropy $H_2(X)$ *of X is defined as*

$$H_2(X) = -\log \mathrm{Col}(X).$$

It should be clear that one can also define $\mathrm{Col}(X|\mathcal{A})$ and $H_2(X|\mathcal{A})$, the collision probability, respectively the Rényi entropy, of X conditioned on the event $\mathcal{A}$, by basing it on the conditional probability distribution $P_{X|\mathcal{A}}$ of X given $\mathcal{A}$.

Definition 14 *Let X and Y be random variables, with respective images $\mathcal{X}$ and $\mathcal{Y}$. Then the* conditional collision probability *and the* conditional Rényi entropy *of X given Y are defined as*

$$\begin{aligned}\mathrm{Col}(X|Y) &= \sum_{y\in\mathcal{Y}} P_Y(y)\cdot \mathrm{Col}(X|Y=y) \quad \textit{and}\\ H_2(X|Y) &= -\log \mathrm{Col}(X|Y)\,,\end{aligned}$$

respectively.

Lemma 7 *Let X, Y be random variables. Then*

$$0 \le H_2(X) \le H(X)\,,$$

and hence

$$0 \le H_2(X|Y) \le H(X|Y)\,.$$

This upper bound on Rényi entropy is an immediate consequence of Jensen's inequality, as can easily be verified. The (conditional) Rényi entropy is maximal, i.e. $H_2(X|Y) = \log|\mathcal{X}|$, if and only if X is uniformly random on $\mathcal{X}$ (and independent of Y). The following shows that if the (conditional) Rényi entropy is close to maximal, and as such $\mathrm{Col}(X|Y)$ close to the minimum value $1/|\mathcal{X}|$, then X is close to uniformly random in terms of statistical distance.

Proposition 12 *Let X, Y be random variables, with respective images $\mathcal{X}$ and $\mathcal{Y}$, and let U be uniformly distributed over $\mathcal{X}$, independent of X and Y. Then*

$$\Delta[XY;UY] \le \frac{1}{2}\sqrt{|\mathcal{X}|\mathrm{Col}(X|Y) - 1}\,.$$

Proof. First, we prove the claim for an "empty" Y. For that, we introduce the 2-distance

$$\Delta_2[X;U] = \sqrt{\sum_{x\in\mathcal{X}} (P_X(x) - P_U(x))^2}$$

and note that (independent of the uniformity of U)

$$\begin{aligned}\Delta[X;U] = \frac{1}{2}\sum_x |P_X(x) - P_U(x)| &= \frac{1}{2}|\mathcal{X}|\frac{1}{|\mathcal{X}|}\sum_x \sqrt{(P_X(x) - P_U(x))^2}\\ &\le \frac{1}{2}|\mathcal{X}|\sqrt{\tfrac{1}{|\mathcal{X}|}\textstyle\sum_x (P_X(x) - P_U(x))^2} = \frac{1}{2}\sqrt{|\mathcal{X}|}\cdot\Delta_2[X;U]\,,\end{aligned}$$

where the inequality follows from Jensen's inequality (with the $1/|\mathcal{X}|$'s as uniform weights/probabilities). Furthermore (now using the uniformity of U),

$$\Delta_2[X;U]^2 = \sum_x \big(P_X(x) - P_U(x)\big)^2 = \sum_x \big(P_X(x)^2 - 2P_X(x)\tfrac{1}{|\mathcal{X}|} + \tfrac{1}{|\mathcal{X}|^2}\big)$$
$$= \sum_x P_X(x)^2 - \frac{1}{|\mathcal{X}|} = \mathrm{Col}(X) - \frac{1}{|\mathcal{X}|}\,.$$

Substituting this into the above gives the claim for an "empty" Y.

The prove for the general claim now follows from Lemma 6 and Jensen's inequality:

$$\Delta[XY;UY] = \sum_y P_Y(y)\Delta[X;U|Y{=}y] \le \frac{1}{2}\sum_y P_Y(y)\sqrt{|\mathcal{X}|\mathrm{Col}(X|Y{=}y){-}1} \le \frac{1}{2}\sqrt{|\mathcal{X}|\mathrm{Col}(X|Y){-}1}\,.$$

This concludes the proof. □

9.2 Universal Hash Functions

An *universal hash function family* consists of a family $\mathcal{G}$ of functions $g : \mathcal{X} \longrightarrow \mathcal{R}$ where $\mathcal{X}$ and $\mathcal{R}$ are fixed non-empty finite sets. Let G be a random variable with uniform probability distribution on $\mathcal{G}$. By applying the function $eval_x : \mathcal{G} \to \mathcal{R}$, $g \mapsto g(x)$ to G, we obtain a new random variable $eval_x(G)$, denoted by $G(x)$, with image $\mathcal{R}$; $G(x)$ describes the value $g(x)$ obtained by choosing $g \in \mathcal{G}$ at random and applying it to x. For the function family to be universal, it is required that for any $x \neq x' \in \mathcal{X}$:

$$P[G(x) = G(x')] \le \frac{1}{|\mathcal{R}|}.$$

There are many constructions of such function families. Here are two examples of universal hash function families that map n-bit strings to r-bit strings, i.e. $\mathcal{X} = \{0,1\}^n$ and $\mathcal{R} = \{0,1\}^r$. In the first example we simply take all the linear functions (described by matrices with r rows and n columns): $\mathcal{G} = \{x \mapsto Ax \,|\, A \in \{0,1\}^{r\times n}\}$. Note that for any $x \neq x' \in \{0,1\}^n$, the function $eval_{x-x'} : \{0,1\}^{r\times n} \to \{0,1\}^r$, $A \mapsto Ax - Ax' = A(x-x')$ is linear and surjective, and as such the cardinality $|eval^{-1}_{x-x'}(\{y\})|$ of the pre-image of $y \in \{0,1\}^r$ is the same for any $y \in \{0,1\}^r$. Therefore, for a random matrix A, $eval_{x-x'}(A)$ is uniformly distributed over $\{0,1\}^r$, and thus the probability that $eval_{x-x'}(A) = 0$, i.e. $Ax = Ax'$, equals 2^{-r}.

In the other example, $\mathcal{X} = \{0,1\}^n$ is identified with the finite field $\mathbb{F}_{2^n}$ by fixing a $\mathbb{F}_2$-basis, and $\mathcal{G}$ is then given by the functions $\{x \mapsto [a \cdot x]_r \,|\, a \in \mathbb{F}_{2^n}\}$, where the multiplication is to be understood as multiplication in $\mathbb{F}_{2^n}$, and $[\,\cdot\,]_r$ denotes the projection into the first r coordinates. One can use a similar argument as above to show universality.

9.3 Extracting Rényi Entropy as Almost Uniform Bits

Theorem 8 (Leftover Hash-Lemma) *Let X be a random variable with image $\mathcal{X}$, and let G be a random variable corresponding to a uniformly random choice of a member of a family of universal hash functions $\mathcal{X} \longrightarrow \{0,1\}^r$. Define $K = G(X)$. Then*

$$H(K|G) \ge H_2(K|G) \ge r - \log(1 + 2^{r-H_2(X)}) \ge r - \frac{2^{r-H_2(X)}}{\ln 2}.$$

Proof. It is sufficient to show the second and third inequalities; the first follows by a previous lemma. We have

$$H_2(G(X)|G) = -\log \mathrm{Col}(G(X)|G) = -\log\Big(\sum_g P_G(g)\cdot \mathrm{Col}(G(X)|G{=}g)\Big)\,,$$

where the sum in the last inequality corresponds to the probability that $g(x_1) = g(x_2)$, where g is uniformly chosen from the function family, and x_1 and x_2 are chosen according to the distribution

of X from $\mathcal{X}$. All choices are made independently. So let us introduce random variables X_1 and X_2, distributed as X, but independently, also from G. Then

$$\begin{aligned}\mathrm{Col}(G(X)|G) &= P[G(X_1)=G(X_2)] \\ &= P[X_1=X_2] + P[X_1\neq X_2]\cdot P[G(X_1)=G(X_2)|X_1\neq X_2] \\ &\leq \mathrm{Col}(X) + (1-\mathrm{Col}(X))\cdot 2^{-r} < 2^{-H_2(X)} + 2^{-r} = 2^{-r}(1+2^{r-H_2(X)})\,,\end{aligned}$$

where the first inequality above follows from the definition of universal hash functions. The second and third inequalities in the statement of the theorem now follow by taking logarithms and using that $\log(1+z)\leq z/\ln 2$. □

Applying Proposition 12 to $\mathrm{Col}(K|G)\leq 2^{-r}\cdot(1+2^{r-H_2(X)})$, we immediately obtain

Corollary 6 *For X, G and K as in Theorem 8,*

$$\Delta[KG;UG]\leq \frac{1}{2}\cdot 2^{-\frac{1}{2}(H_2(X)-r)}\,,$$

where U is uniformly distributed over $\{0,1\}^r$ (independent of anything else).

9.4 Privacy Amplification

An immediate consequence is that if the Rényi entropy about the original string is large enough from the point of view of an adversary who has access to partial information, then we can distill a shorter key that is almost random, and about which the adversary has almost no information.

Theorem 9 (Privacy Amplification) *Let X and Y be random variables with respective images $\mathcal{X}$ and $\mathcal{Y}$, and let G be a random variable corresponding to a uniformly random choice of a member of a family of universal hash functions $\mathcal{X}\longrightarrow\{0,1\}^r$. Define $K=G(X)$. Then*

$$\Delta[KYG;UYG]\leq \frac{1}{2}\cdot 2^{-\frac{1}{2}(H_2(X|Y)-r)}\,,$$

where U bis uniformly distributed over $\{0,1\}^r$ (independent of anything else).

Suppose that $H_2(X|Y)\gg r$. Then the theorem implies that for an adversary with side information Y, the key K is as good as uniform and unpredictable from the point of view of the adversary, except with small probability.

Proof. The proof follows immediately from Corollary 6 and Jensen's inequality:

$$\begin{aligned}\Delta[KYG;UYG] &= \sum_y P_Y(y)\Delta[KG;UG\,|Y=y] \leq \frac{1}{2}\sum_y P_Y(y)\cdot 2^{-\frac{1}{2}(H_2(X|Y=y)-r)} \\ &= \frac{1}{2}\sum_y P_Y(y)\sqrt{\mathrm{Col}(X|Y=y)}\cdot 2^{\frac{r}{2}} \leq \frac{1}{2}\sqrt{\textstyle\sum_y P_Y(y)\mathrm{Col}(X|Y=y)}\cdot 2^{\frac{r}{2}} \\ &= \frac{1}{2}\sqrt{\mathrm{Col}(X|Y)}\cdot 2^{\frac{r}{2}} = \frac{1}{2}\cdot 2^{-\frac{1}{2}(H_2(X|Y)-r)}\,.\end{aligned}$$

□

9.5 Application to Eavesdropping

Suppose the adversary has partial information defined by an *eavesdropping function*

$$e:\mathcal{X}\longrightarrow\{0,1\}^t.$$

This means that he has access to the value $y = e(x)$. Possibly, the eavesdropping function has been specified by the adversary, and our only knowledge about it is that it maps to t-bit strings (so we know t).

We apply Privacy Amplification to mod out this partial information. Let notation be as before, with exceptions as stated in the theorem below. The partial information Y is now defined as $e(X)$.

Theorem 10 *Assume that $\mathcal{X} = \{0,1\}^n$, and that X has the uniform distribution on it. Assume $0 \leq t < n$. Let $0 < s < n - t$ be a safety parameter. Define $r = n - t - s$. Then, applying privacy amplification,*

$$\Delta(KGY; UGY) \leq 2^{-\frac{s}{2}-1}.$$

Proof. For $y \in \{0,1\}^t$, let c_y denote the number of $x \in \{0,1\}^n$ such that

$$e(x) = y.$$

Since X has the uniform distribution on $\{0,1\}^n$, we have

$$\mathrm{Col}(X|Y{=}y) = c_y \cdot \frac{1}{c_y^2} = \frac{1}{c_y}$$

and therefore

$$\mathrm{Col}(X|Y) = \sum_y P_{e(X)}(y)\mathrm{Col}(X|Y{=}y) = \sum_y \frac{c_y}{2^n}\frac{1}{c_y} = 2^{t-n}.$$

This means that

$$H_2(X|Y) = n - t$$

and therefore by privacy amplification,

$$\Delta(KGY; UGY) \leq \frac{1}{2} \cdot 2^{-\frac{1}{2}(n-t-r)} = \frac{1}{2} \cdot 2^{-\frac{s}{2}}.$$

□

References

[1] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Ueli Maurer. *Generalized Privacy Amplification.* Transactions on Information Theory, vol. 41, no. 6. IEEE, 1995.

[2] Thomas M. Cover and Joy A. Thomas. *Elements of information theory.* 2nd edition. Wiley, 2006. ISBN 0-471-24195-4.

[3] Victor Shoup. *A Computational Introduction to Number Theory and Algebra.* 2nd edition. Cambridge University Press, 2008. ISBN 978-0-521-51644-0.

[4] Stefan Wolf. *Unconditional Security in Cryptography.* Lecture Notes in Computer Science, vol. 1561. Springer, 1999.