ENTROPY AND MUTUAL INFORMATION IN INFORMATION THEORY

ALEXANDROS GELASTOPOULOS

Abstract. We introduce the concepts of information-theoretic entropy and mutual information, emphasizing their common interpretations. We highlight their role in some basic theorems of information theory, namely the asymptotic equipartition property and Shannon's source coding and channel coding theorems.

1. Introduction

Information theory is all about the quantification of information. It was developed by C. Shannon in an influential paper of 1948, in order to answer theoretical questions in telecommunications.

Two central concepts in information theory are those of entropy and mutual information. The former can be interpreted in various ways and is related to concepts with the same name in other fields, including statistical mechanics, topological dynamics and ergodic theory. In information theory, entropy is a measure of the randomness of a discrete random variable. It can also be thought of as the uncertainty about the outcome of an experiment, or the rate of information generation by performing the experiment repeatedly.

Mutual information is the information that one random variable contains about another random variable. It is defined in terms of entropy and conditional entropy. The importance of both entropy and mutual information can be seen through their appearance in several important theorems of information theory, although their applications extend to other fields. Our goal here is to give a motivated introduction of entropy and mutual information and to present some of the highlights of their role in information theory. In particular, we discuss the Asymptotic Equipartition Property, Shannon's source coding theorem and Shannon's channel coding theorem. Most of our discussion is based on [1].

Sections 2-5 deal with entropy only. Section 2 serves as a motivation, while section 3 introduces entropy and its basic properties. Section 4 is about the Asymptotic Equipartition Property, while section 5 discusses the source coding problem. Conditional entropy and mutual information are introduced in section 6. Some insightful properties of mutual information are discussed in section 7, including a relation to the concept of sufficient statistics. Section 8 discusses the channel coding problem, while section 9 makes the connection between source and channel coding.

Date: April 24, 2014.

This is the second project for the class Probability Theory 2, taught by Murad Taqqu in Spring 2014, at Boston University. It is an extension of the first project *Information-theoretic Entropy*.

2. Motivation: Measuring information

Consider the uniform distribution on the unit square and let $A_1, \ldots, A_5$ be the partition shown in Figure 1. Suppose you perform an experiment with this random variable and you want to describe in which part of the square the random variable took its value. If the event A_1 occurs, then you can describe it by saying it is on the right half of the square. Since there are only two halfs, this is one piece of binary information; it is one bit of information.

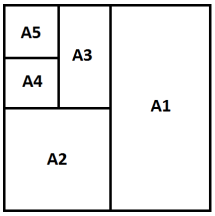

Figure 1. Assuming uniform distribution, the probability of each event is proportional to its area.

If the result is A_2, then you need two bits of information: it is on the left, in contrast to the right, and it is on the bottom half, in contrast to the top half. Similarly, if you were to describe A_4 in this way, you would need 4 bits of information. Notice that as the sets get smaller, we need more bits to describe the event, but the amount of precision we get is also higher.

In general, an event with probability 2^{-k} requires k bits of information to be described. Turning this around, an event with probability $P(A)$ requires $-\log P(A)$ bits to be described (in this text, all logarithms are base 2).

Motivated by this example, we define the **self-information** of the event A by

$$I(A) = -\log P(A).$$

This function has two important properties. First, it is a decreasing function of $P(A)$; events with smaller probability contain more information. Second, if A and B are independent, then

$$I(A \cap B) = I(A) + I(B).$$

In words, the self-information of the intersection of two independent events is the sum of the self-informations.

3. Definition of entropy

We now restrict ourselves to random variables with finitely many possible values (although many definitions can be extended to the countably infinite case).

Let X denote a random variable with values in the finite set $\mathcal{X} = \{x_1, \ldots, x_n\}$ and denote by p_i the probability of the outcome x_i. According to the above definition, the self-information of x_i is $-\log p_i$. We define the **entropy** H of X by

$$H(X) := -\sum_{i=1}^{n} p_i \log p_i.$$

(Here we use the convention that $0 \cdot \log 0 = 0$.)

Notice that $H(X)$ is the expected value of the self-information. To make this more precise, let A_X denote the simple event in which X takes its value. That is, $A_X = \{x_i\}$ whenever $X = x_i$. The quantity $-\log P(A_X)$ is then a random variable and $H(X)$ is by definition its expectation. According to the above intuitive approach to the self-information, the entropy describes the information we get on average by performing an experiment and observing the value of X.

Here are some of the properties of entropy.

(1) $H(X) \geq 0$, with equality if and only if X is deterministic, that is $p_i = 1$ for some i.
(2) $H(X) \leq \log |\mathcal{X}|$, with equality if and only if X is uniformly distributed.
(3) $H((X,Y)) \leq H(X) + H(Y)$, with equality if and only if X and Y are independent.

Remark 3.1. According to the second property, the entropy $H(X)$ is maximized when the distribution is uniform. In some sense, the uniform distribution has the highest randomness; we cannot make any meaningful prediction about its outcome before the experiment. On the other hand the entropy is zero when X is purely deterministic. The above justifies thinking of $H(X)$ as the *randomness* of X. Stated in a different way, $H(X)$ is the *uncertainty* we have about the outcome of X before we observe it.

Remark 3.2. The above interpretation of entropy is also supported by the third property: if X and Y are independent, the uncertainty about the vector (X,Y) is the sum of the uncertainties. Otherwise, the joint uncertainty is smaller, because by observing X we get some information about Y. Notice that this is reminiscent of the behavior of the variance for a sum of two random variables $X + Y$, but here it is about the random vector (X,Y). Another difference is that the relation $\sigma^2_{X+Y} = \sigma^2_X + \sigma^2_Y$ holds if X and Y are merely *uncorrelated*, while $H((X,Y)) = H(X) + H(Y)$ means that X and Y are *independent*.

4. The Asymptotic Equipartition Property

We now proceed to see a theorem where $H(X)$ plays a central role. Suppose for simplicity that X is a Bernoulli random variable, taking value 1 with probability p and 0 otherwise. Its entropy is a number between 0 and 1. More precisely, it is given by

$$H(X) = -p \log p - (1-p) \log(1-p).$$

Let $X_1, X_2, \ldots$ be a sequence of i.i.d. random variables, each with the same distribution as X. For a fixed n, the random vector $(X_1, \ldots, X_n)$ takes values that are strings of 0's and 1's. The probability of such a string depends on the number of 1's that it contains. More precisely if the string $x_1 x_2 \ldots x_n$ contains k ones, then

$$P(x_1 x_2 \ldots x_n) = p^k \cdot (1-p)^{n-k}. \tag{4.1}$$

What does a *typical* string look like? Of course, each string has a non-zero probability of occurring, if $0 < p < 1$. But as n gets large, can we say something about the form of a string with high probability? It turns out that something can be said about the number of 1's and 0's in a typical string and this is the content of the Asymptotic Equipartition Property (AEP).

Notice that $P(X_1 X_2 \ldots X_n)$ is itself a random variable: it is the probability of the observed string. By Eq. 4.1, it is a function of the number of 1's and it is a

one-to-one function if $p \neq \frac{1}{2}$. We will state the theorem for an arbitrary random variable with finitely many possible values, not necessarily Bernoulli. In this case, $P(X_1X_2 \dots X_n)$ is a function of the number of occurrences of each symbol.

Theorem 4.1 (Asymptotic Equipartition Property)**.** *If $X, X_1, X_2, \dots$ are i.i.d. random variables with finitely many possible values and $P(X_1X_2 \dots X_n)$ denotes the probability of the random string $X_1X_2 \dots X_n$, then*

$$-\frac{1}{n} \log P(X_1X_2 \dots X_n) \to H(X) \ \ a.s.$$

The proof of the above theorem is a direct application of the strong law of large numbers to the independent random variables $-\log P(X_i)$. But its consequences are striking. As already mentioned, $P(X_1X_2 \dots X_n)$ is a function of the number of occurrences of each symbol. The above theorem hence tells us something about the asymptotic behavior of the constitution of the string or, in other words, how many of each symbol appear as n gets large.

To get some more detailed results we define, for a given $\epsilon > 0$, the *typical set* $A_\epsilon^{(n)}$ as the set of strings $x_1x_2 \dots x_n$ such that

$$2^{-n(H(X)+\epsilon)} \leq P(x_1x_2 \dots x_n) \leq 2^{-n(H(X)-\epsilon)},$$

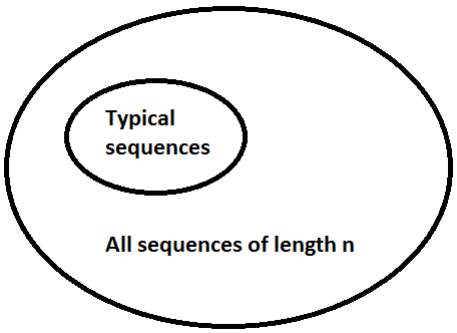

FIGURE 2. There are $2^{n \log |\mathcal{X}|}$ sequences of length n, but only about $2^{nH(X)}$ typical sequences.

Notice that the above inequalities are equivalent to

$$H(X) - \epsilon \leq -\frac{1}{n} \log P(x_1x_2 \dots x_n) \leq H(X) + \epsilon,$$

where the quantity in the middle is the one appearing in the Asymptotic Equipartition Property. The following theorem, which is a consequence of the AEP, justifies the name "typical set" (see [1] for a proof).

Theorem 4.2.

(1) $\lim_{n\to\infty} P\left(A_\epsilon^{(n)}\right) = 1$

(2) $\left|A_\epsilon^{(n)}\right| \leq 2^{n(H(X)+\epsilon)}$

(3) $\lim_{n\to\infty} \left|A_\epsilon^{(n)}\right| \geq 2^{n(H(X)-\epsilon)}$

The first property states that the probability that a non-typical sequence will occur goes to 0 for large n. The last two properties state that there are not too many typical sequences; they are of order $2^{nH(X)\pm\epsilon}$. This should be compared to the total number of possible sequences $2^{n\log|\mathcal{X}|}$ and recall that $H(X) < \log|\mathcal{X}|$, unless X is uniformly distributed. That is, the number of typical sequences is *exponentially* smaller than the total number of sequences.

The three properties combined say that the probability distribution is "concentrated" on this small set, of order roughly $2^{nH(X)}$. By the definition of $A_\epsilon^{(n)}$, each of these typical sequences has roughly the same probability. All in all, the probability distribution of the sequences of length n is roughly a uniform distribution on a set with about $2^{nH(X)}$ elements. Stated in another way, if you randomly generate a sequence of length n from the random variable X, it is almost the same as uniformly picking a sequence from the much smaller set of typical sequences. The size of this set is determined by the entropy of X.

5. Source coding

In digital telecommunications, the following setup is typical: You have a random source that at each instance produces symbols from a finite set and you want to encode these symbols into binary strings, that is to assign a unique string of 0's and 1's to each symbol, before you trasmit them digitally (see Figure 3 and Table 1). These strings are called codewords and the goal is to choose them in a way that minimizes the expected number of bits you transmit.

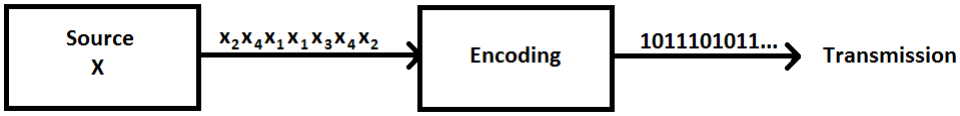

Figure 3. The symbols produced by a source are encoded into strings of 0's and 1's, before they are transmitted through a digital medium.

Symbol	Codeword
x_1	1001
x_2	01101
x_3	101
...	...
x_N	0101

Table 1. To each symbol we assign a unique string of 0's and 1's.

The source is modelled by a random variable X that takes values in the finite set $\mathcal{X} = \{x_1, \dots x_N\}$. The problem of source coding consists of finding a one-to-one function

$$f : \mathcal{X} \to \{\text{finite strings of 0's and 1's}\},$$

with some extra restrictions which we discuss promptly.

Since we want to minimize the bits that we transmit, we would like to use codewords of small length, starting with length 1. But assuming that we transmit more than one codewords, one after the other, and the receiver of our message wants to decode it, they must be able to identify where each codeword ends. So the restriction we impose is that *no codeword is the beginning of another codeword.* For example, if the string '11' is used as a codeword, then '1101' cannot be used. The following example compares two ways to encode a source.

Example 5.1. Let $\mathcal{X} = \{x_1, x_2, x_3, x_4\}$ with probabilities

$$p_1 = \frac{1}{2}, \quad p_2 = \frac{1}{4} \quad \text{and } p_3 = p_4 = \frac{1}{8}.$$

A naive way to encode this source would be to use codewords of length 2, that is '00', '01', '10' amd '11'. Then the expected length of a codeword is also 2.

But we could actually do better if we took advantage of the fact that x_1 appears much more often than the rest of the symbols. We therefore assign to x_1 the shorter codeword '1'. Now, no other codeword starting with '1' can be used. Therefore, we choose to use the codeword '01' for x_2. Since no other codeword can start with either '1' or '01', we are forced to choose '000' and '001' for x_3 and x_4. Comparing this to the previous case where each codeword had length 2, we notice that some keywords now are shorter and some are longer. But what happens on average? If we denote by $l(x_i)$ the length of the codeword for x_i, we have

$$\begin{aligned} E[l(X)] &= \frac{1}{2}l(x_1) + \frac{1}{4}l(x_2) + \frac{1}{8}l(x_3) + \frac{1}{8}l(x_4) \\ &= \frac{1}{2} + \frac{1}{2} + \frac{3}{8} + \frac{3}{8} = \frac{7}{4} < 2. \end{aligned}$$

We conclude that this way of encoding is more efficient.

A natural question to ask is *how good can we do?* In other words, what is the shortest expected length that we could hope to achieve? It turns out that a lower bound for the expected length of a source is the entropy $H(X)$. In the above example, this bound is actually achieved by the second coding scheme, since $H(X) = \frac{7}{4}$. But can it always be achieved?

The simple answer is no, but the reason is the discrete nature of the length of a keyword. Ideally, we would like to assign a codeword of length $-\log p_i$ to the symbol x_i. In the above example, this worked well, because $-\log p_i$ was an integer for each i. In general this will not be the case and the remedy will be to use $\lceil -\log p_i \rceil$ number of bits, where $\lceil r \rceil$ denotes the smallest integer at least as big as r. This would increase the average length by at most 1 bit. This is captured in the following proposition.

Theorem 5.2. *Let X be a random source and suppose that L is the minimum average codeword length over all possible codes. Then*

$$H(X) \leq L < H(X) + 1.$$

If $L > H(X)$ and we are not satisfied with the difference $L - H(X)$, we can improve our average length by combining symbols. That is, if we want to transmit a large number of symbols from the set $\mathcal{X}$, we can group them into n-tuples, thus effectively using $Y = (X_1, \ldots, X_n)$ as our new random variable and $\mathcal{Y} = \mathcal{X}^n$ as our new set of symbols (see Table 2).

Symbol pair	Codeword
x_1x_1	1001
x_1x_2	01101
...	...
x_1x_N	11011
x_2x_1	0001
x_2x_2	101
...	...
x_Nx_N	001

TABLE 2. Instead of assigning codewords to single symbols, we combine symbols into blocks of length n. Here $n = 2$.

The entropy of the vector $Y = (X_1, \ldots, X_n)$ is $nH(X)$, since $X_1, \ldots, X_n$ are independent. By the above theorem, the minimum expected length of a codeword for Y is between $nH(X)$ and $nH(X)+1$. Therefore, the minimum expected length *per symbol of* X will be in the range $\left[H(X), H(X) + \frac{1}{n}\right)$. We have therefore proved the following theorem.

Theorem 5.3. *Let X be a random source and suppose that in the source coding problem we combine symbols into blocks of length n. If L_n is the per symbol minimum average codeword length, then*

$$H(X) \leq L_n < H(X) + \frac{1}{n}.$$

As a corollary we get Shannon's source coding theorem ([3]).

Corollary 5.4 (Shannon's source coding theorem)**.** *For any $\epsilon > 0$, there exists a code with per symbol average codeword length $L < H(X)+\epsilon$, provided that we allow for arbitrarily large block length. No code exists with $L < H(X)$.*

The source coding theorem states that $H(X)$ is not only a lower bound, but also an asymptotic upper bound for the average number of bits per symbol that are needed to encode a long message from X. This suggests the interpretation of $H(X)$ as the rate of information generation of the source X ([3]).

Another way to view $H(X)$ is as follows: while X can produce $|\mathcal{X}|$ different symbols, the number of bits needed to encode it is $H(X)$, same as a source with only $2^{H(X)}$ symbols. That is, the "effective" number of symbols of X is $2^{H(X)}$.

6. Conditional Entropy and Mutual Information

Given two random variables X and Y, taking values in the finite sets $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_m\}$, respectively, the **joint entropy** of X and Y is defined as the entropy of the vector (X, Y). That is, if $p_{ij} = P(X = x_i, Y = y_j)$, then

$$H(X, Y) = -\sum_{i,j} p_{ij} \log p_{ij}.$$

In dealing with two random variables, it is useful to introduce the notation

$$p_{X,Y}(x, y) := P(X = x, Y = y)$$

for the joint probability of X and Y and

$$\begin{aligned} p_X(x) &:= P(X = x) \\ &= \sum_{y \in \mathcal{Y}} P(X = x, Y = y) \end{aligned}$$

for the marginal probability of X, and similarly for Y.

The *a posteriori* probability $p_{X|Y}(x|y)$ is defined to be the conditional probability of $X = x$, given that $Y = y$, that is

$$p_{X|Y}(x|y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}.$$

(Here we use the convention $\frac{0}{0} = 0$.)

Intuitively, if we know that $Y = y$, then the distribution of X "changes" from $p_X(\cdot)$ to $p_{X|Y}(\cdot|y)$. We denote by $H(X|Y = y)$ the entropy of the new distribution of X, that is

$$H(X|Y = y) = -\sum_{i=1}^{n} p_{X|Y}(x|y) \cdot \log p_{X|Y}(x|y).$$

This can be thought of as the entropy of X knowing that $Y = y$.

The **conditional entropy** of X given Y, denoted by $H(X|Y)$, is the expectation of the above quantity (which depends only on y). That is (with a slight abuse of notation),

$$\begin{aligned} H(X|Y) &:= EH(X|Y) \\ &= \sum_{y \in \mathcal{Y}} p_Y(y) \cdot H(X|Y = y) \\ &= -\sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p_Y(y) \cdot p_{X|Y}(x|y) \cdot \log p_{X|Y}(x|y) \\ &= -\sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p_{X,Y}(x, y) \cdot \log p_{X|Y}(x|y) \end{aligned}$$

We can interpret this as the expected *new* entropy of X, *after we observe* Y.

There is a more direct, but perhaps less intuitive way to define the conditional entropy. Recall that A_X denotes the simple event in which X takes its value, that is $A_X = \{x\}$ when $X = x$. Then

$$H(X|Y) = E\left[-\log P(A_X|Y)\right],$$

which can be compared to $H(X) = E\left[-\log P(A_X)\right]$.

Conditional entropy satisfies the following very intuitive properties (see [1]).

(1) $H(X|Y) \leq H(X)$ with equality if and only if X, Y are independent.
(2) $H(X|Y) \geq 0$, with equality if and only if X is completely determined by Y (that is, $X = f(Y)$ a.s., for some function $f : \mathcal{X} \to \mathcal{Y}$).
(3) $H(X, Y) = H(Y) + H(X|Y)$

By recalling our interpretation of entropy as the uncertainty about the outcome of an experiment, the first property says that the uncertainty we have about X cannot increase (on average) by observing Y. This justifies thinking of $H(X|Y)$ as the *remaining* uncertainty of X, after observing Y. The second property says that

the remaining uncertainty cannot be negative, but it is zero if by observing Y we have all the information about X.

The third property says that the uncertainty about the joint outcome of X and Y can be broken into the uncertainty about Y plus the remaining uncertainty of X after observing Y.

By the symmetry of $H(X,Y)$, the second property implies the relation

$$H(Y) + H(X|Y) = H(X) + H(Y|X). \tag{6.1}$$

From here it is clear that in general $H(X|Y) \neq H(Y|X)$.

Consistent with the interpretation of $H(X)$ and $H(X|Y)$ as the uncertainty of X *before* and *after* observing Y, respectively, we can think of the difference $H(X) - H(X|Y)$ as the *information* we gain about X by observing Y. Notice that by Eq. 6.1 we have the symmetry relation $H(X) - H(X|Y) = H(Y) - H(Y|X)$, which motivates calling this quantity the *mutual information* of X and Y. We denote this by $I(X;Y)$, that is

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

From the properties of entropy and conditional entropy, it is easy to derive the following:

(1) $I(X;Y) \geq 0$, with equality if and only if X and Y are independent.
(2) $I(X;Y) \leq \min\{H(X), H(Y)\}$ with equality if and only if X is completely determined by Y or vice versa.

The above imply that $I(X,Y)$ is a measure of the non-independence of X and Y or, in other words, a measure of the extent to which one of them determines the other. This observation will be further explored in the next section.

7. Data-processing inequality and sufficient statistics

If $I(X,Y)$ is the information we extract about X by observing Y, then one would expect that we cannot increase this information by simply transforming Y. That is, if $Z = f(Y)$, where f is a deterministic function, we expect that $I(X,Z) \leq I(X,Y)$. This is indeed true and it is a special case of the next theorem.

We say that X and Z are **conditionally independent given** Y if for all x, y, z,

$$p_{X,Z|Y}(x,z|y) = p_{X|Y}(x|y) \cdot p_{Z|Y}(z|y).$$

Multiplying both sides by $p(y)$ we get $p_{X,Y,Z}(x,y,z) = p_{X,Y}(x,y) \cdot p_{Z|Y}(z|y)$, hence conditional independence of X and Z given Y is equivalent to the sequence X, Y, Z being a Markov chain in this order.

The above is also equivalent to say that $Z = f(Y, \theta)$, where θ is a random variable independent of X, uniformly distributed in $[0,1]$ (see Theorem 6.13 in [2]). In particular, it includes the case $Z = f(Y)$ considered above. We have the following theorem (Theorem 2.8.1 in [1]):

Theorem 7.1. *If X and Z are conditionally independent given Y, then $I(X,Z) \leq I(X,Y)$.*

This theorem is sometimes stated as follows: We cannot improve the inferences that can be made from the data by manipulating the data ([1]).

But what about *retaining* the inferences that can be made, while reducing the data? Recall that a sufficient statistic for the family of distributions $\{p_{X|Y}(\cdot|x)\}_x$

is a statistic $g(Y)$ such that $P(Y = y|g(Y) = t, X = x) = P(Y = y|g(Y) = t)$ for all x, y, t (in general X doesn't need to be a random variable, but merely a parameter). The following theorem states that sufficiency is exactly what is needed for a statistic to preserve the information ([1], section 2.9).

Theorem 7.2. *Let $g(Y)$ be a statistic of Y. We have*

$$I(X, Y) = I(X, g(Y))$$

if and only if $g(Y)$ is sufficient for the family of distributions $\{p_Y(\cdot|X = x)\}_x$ for any distribution on X.

8. Channel coding

We now turn to another basic theorem of information theory: the channel coding theorem. The context of the problem is the following: We want to transmit an array of symbols through a noisy communication channel. That is, we send a symbol from one end of the channel and we receive a noisy version of the symbol at the other end of the channel. The goal is to infer what symbol was sent by observing the received symbol only. Here we focus on discrete channels, that is channels for which there is a finite number of possible symbols both at the sender and the receiver end.

The setup is shown in Fig. 4. The main component is a communication channel, which consists of an *input alphabet* $\mathcal{X}$, an *output alphabet* $\mathcal{Y}$ and a collection $\{P^x\}_x$ of distributions on $\mathcal{Y}$, one for each element $x \in \mathcal{X}$. The central idea here is that for each input $x \in \mathcal{X}$, we don't get a deterministic output $y \in \mathcal{Y}$, but a *probability distribution* on $\mathcal{Y}$, which can be thought of as the result of noise.

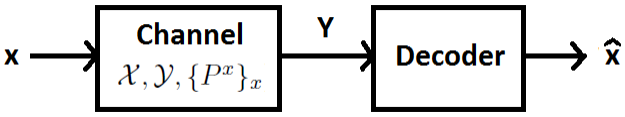

Figure 4. The channel coding problem.

The sender wants to transmit an element from the input alphabet, which we denote by x. Once it is inserted into the channel, another element $y \in \mathcal{Y}$ comes out at the other end of the channel. This output is random, with distribution P^x. Then a decoder takes y and maps it into an element $\widehat{x}$, which ideally is the same as x. The problem for the decoder is that different x's might produce the same y, so there is no way to know with certainty how to decode each y. The decision has to be made purely on probabilistic terms, so that to minimize the chance of error. We assume that the probability distributions P^x are known. If the distribution of the input X is also agreed upon, then the optimal decoder is clearly a Bayesian one. The channel coding problem roughly amounts to choosing the distribution of the input, so that the probability of error is small, while maximizing the rate of information transfer (to be defined).

For example, consider the noisy binary channel of Fig. 5. Here the sender inputs either 0 or 1 and the receiver observes the correct symbol with probability $1 - \epsilon$. Assuming that ϵ is small, then the receiver will decode a 0 as a 0 and a 1 as a 1. This way the probability of error is ϵ, independently of the input.

Sometimes using fewer symbols from the input alphabet might allow the receiver to decode the output with certainty. Consider for example the channel of Fig. 6. If

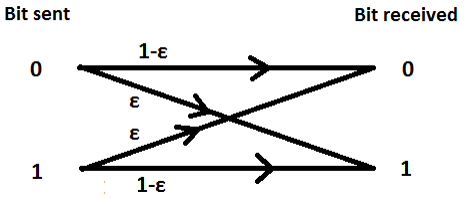

FIGURE 5. The noisy binary channel. For each bit sent, there is $1 - \epsilon$ probability that it will be transmitted correctly.

all four symbols are into play, then the receiver won't be able to decode the output with certainty. But if the sender and the receiver agree that only a and c are used from the input alphabet, then decoding is possible: a or b means a, while c or d means c. Therefore, by reducing the available symbols to half, we have achieved perfect decodability. Look at the price we had to pay, though: now the sender can use only two symbols to communicate a message to the receiver, which means that the messages will become longer.

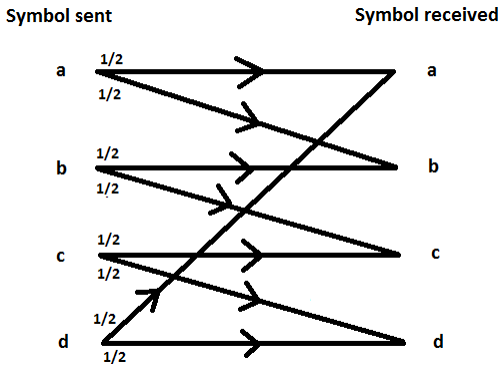

FIGURE 6. For each symbol sent, the symbol received is either the same or the next one, with probability 1/2 each.

The above trick does not always work, as in the noisy binary channel of Fig. 5 (of course it doesn't make sense to have only one possible symbol). Is there something we can do here? As one might expect, there is no way to achieve perfect decodability using such a channel. But one might hope to *reduce* the probability of error by using the channel repeatedly. For example, suppose that in the case of the noisy binary channel the sender and receiver agree that the sender will repeat each symbol k times. Due to noise, sometimes the receiver will see the wrong symbol, but if ϵ is small, it is very unlikely that the wrong symbol will appear more than half of the times. Therefore, the receiver can decode based on which symbol appeared most of the times, hoping that this is the actual symbol sent. By the law of large numbers, the probability of error will tend to 0 as k increases. The trade-off is that the rate of transmission becomes smaller. Instead of sending one novel symbol per use of the channel, we are transmitting one symbol per k uses, hence reducing the rate of information transfer k-fold.

Although we still haven't given a precise definition of the rate, in the above example it is clear that in order to get an arbitrarily low probability of error, the rate will also become arbitrarily small. What we would like would be to achieve an arbitrarily small error with a constant, non-zero rate. It turns out that this is possible, under one assumption: the message that we want to transmit is big enough, so that we can combine symbols into blocks, like we did in the problem of source coding. This is the content of the channel coding theorem. But before we state it we need some definitions.

A **discrete channel** is a triplet $(\mathcal{X}, \mathcal{Y}, \{P^x\}_x)$, where $\mathcal{X}$ and $\mathcal{Y}$ are two finite sets and P^x is a probability distribution on $\mathcal{Y}$ for each $x \in \mathcal{X}$. The distribution P^x can be thought of as the conditional distribution of the output, when the input is x.

More precisely, if we consider the input X as a random variable taking values in $\mathcal{X}$, then the output also turns into a random variable Y. The joint distribution $P_{X,Y}$ of (X, Y) is induced by the distribution P_X of the input and the channel distributions P^x as follows:

$$P_{X,Y}(x, y) = P_X(x) \cdot P^x(y). \tag{8.1}$$

The fact that $P^x(y)$ behaves like a conditional distribution is seen by comparing the above equation to $P_{X,Y}(x, y) = P_X(x) \cdot P_{Y|X}(y|x)$. The fine distinction here is that by looking at the channel only, we don't have any "joint distribution"; in fact, the definition of the channel does not even treat the input X as a random variable. But once we choose a distribution for X and turn it into a random variable, we immediately get a joint distribution $P_{X,Y}$.

This allows one to choose the distribution for X in an optimal way, so that to facilitate the communication. Since our goal is to infer X by observing Y, we might want to choose the distribution so that their mutual information (which is determined by the joint distribution) is maximized. Indeed, we will see that the ultimate quality of the channel is determined by $\max_{P_X} I(X, Y)$, where P_X ranges over all possible distributions on X.

The k-th extension of the channel $(\mathcal{X}, \mathcal{Y}, \{P^x\}_x)$ is the channel with input alphabet $\mathcal{X}^k$, output alphabet $\mathcal{Y}^k$, and $P^{x_{i_1}, \dots, x_{i_k}}$ equal to the product of the distributions $P^{x_{i_1}}, \dots, P^{x_{i_k}}$. Intuitively, this corresponds to treating k subsequent uses of the channel as one, since the input is an element in $\mathcal{X}^k$ and the output is an element in $\mathcal{Y}^k$. We note that the use of the product distribution implies that each output symbol depends only on the respective input symbol and not on the past (or future) ones. In other words, we are considering a **memoryless channel**.

An (M, k) code for the channel $(\mathcal{X}, \mathcal{Y}, \{P^x\}_x)$ is a way to choose M elements in the set $\mathcal{X}^k$, together with a rule to decide which one was sent by observing the output only. More precisely, an (M, k) code consists of the following:

- A subset $A \subset \mathcal{X}^k$ of cardinality $|A| = M$.
- A decoding function $g : \mathcal{Y}^k \to A$.

The set A is the subset of $\mathcal{X}^k$ that we will be actually using. This is the same idea as in the channel of Fig. 6, where we used only 2 out of the 4 symbols of $\mathcal{X}$. Consistent with the convention that 2^n symbols carry n bits of information, we define the **rate** of an (M, k) code to be $\frac{\log M}{k}$, measured in *bits per channel use*.

For each $x \in A$, the **probability of error** is

$$\lambda_x = P^x \left(g(Y) \neq x \right).$$

In other words, given that the input is x, we consider the (conditional) distribution P^x of the output variable Y and we look at the probability that this output will be decoded wrongly, that is $g(Y) \neq x$.

The **maximal probability of error** is $\lambda = \max_{x \in A} \lambda_x$.

We say that R is an **achievable rate** for the channel $(\mathcal{X}, \mathcal{Y}, \{P^x\}_x)$ if for any $\epsilon > 0$, there exists an (M, k) code with rate at least R (that is, $\log M \geq kR$) and maximal probability of error $\lambda < \epsilon$. The **capacity** of a channel is the supremum of all achievable rates.

We are now ready to state the channel coding theorem (Theorem 7.7.1 in [1]):

Theorem 8.1 (Shannon's channel coding theorem)**.** *Let $(\mathcal{X}, \mathcal{Y}, \{P^x\}_x)$ be a discrete memoryless channel. Then, its capacity is equal to $\max_{P_X} I(X, Y)$, where P_X ranges over all possible distributions on $\mathcal{X}$ and (X, Y) has the induced joint distribution given by 8.1.*

The importance of the above theorem lies in the fact that it relates a purely communicational measure (the maximum rate of information transfer) to a purely probabilistic one (the maximum mutual information).

9. Independence of source and channel coding

Let's now look at how the two communications problems that we have considered, namely the source coding and the channel coding, relate to each other.

In the source coding problem, we have a source of symbols with a given distribution and we want to encode these symbols so that the average length is minimized. The channel problem picks up at this point and considers the encoded sequence as the message that needs to be transmitted. Once the message is recovered at the other end of the channel, it is then translated back into the original message, by inverting the source coding procedure. This is summarized in Fig. 7. Notice that there is a two-step decoding process at the receiver end: one that inverts the effect of the channel and one that inverts the effect of the source coding.

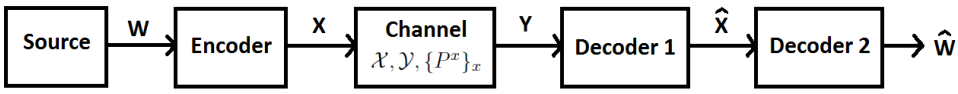

Figure 7. The combined source-channel coding problem.

Note that although in the source coding problem we were assuming that the symbols are mapped into binary words, everything would work the same if we used M-ary words. In fact, if the channel is using an (M, k)-code, then we are forced to use M-ary source coding, if we want to avoid an intermediate step (not shown here). On the other hand, the probability distribution of the M-ary digits produced by the source coding is irrelevant to the channel coding problem, since in the definition of the achievable rates for the channel we are looking at the *maximal* probability of error.

This implies that the two problems are considered independently. Indeed, the channel coding process sees only an M-ary string that has to be transmitted; it doesn't know how it was produced (hence makes no assumption on the probability

distribution of the input). Similarly, the source coding process is interested in minimizing the total length of the message, without caring how it will be transmitted.

It is reasonable then to ask if we could achieve a better result on the composite problem by merging the processes of source and channel coding. That is, we might want to map the symbols of the source into M-ary words, with an eye towards the way that they will be transmitted through the channel. This might lead us to a suboptimal solution for the source coding problem, but it might allow for a better result in the channel coding problem. Then one might hope that in total we will get a better result.

If the average length of a codeword produced by the source coding is L (bits per symbol), while the rate of the channel code is R (bits per channel use), then in total we are transmitting $\frac{R}{L}$ symbols per channel use. Recall that the capacity C of the channel is the supremum of the achievable rates and the entropy $H(Z)$ of the source Z is the infimum of the average codeword length in source coding. That is, by considering the two problems independently we are able to achieve (asymptotically) a rate of $\frac{C}{H(Z)}$ symbols per channel use. The next theorem states that no improvement can be made by combining the two problems (Theorem 7.13.1 in [1]).

Theorem 9.1. *Let Z be a random source of symbols taking values in $\mathcal{Z}$ and let $(\mathcal{X}, \mathcal{Y}, \{P^x\}_x)$ be a discrete memoryless channel. Suppose that for each $\epsilon > 0$ there exists a function $f : \mathcal{Z}^l \to \mathcal{X}^k$ that maps blocks of symbols produced by Z directly into the alphabet of the k-th extension of the channel, with $\frac{l}{k} \geq R$, and a decoding function $g : \mathcal{Y}^k \to \mathcal{Z}^l$, such that the maximal probability of error $\lambda = \max_{x \in f(\mathcal{Z}^l)} \lambda_x$ is less than ϵ. Then, $R \leq \frac{C}{H(Z)}$, where C is the capacity of the channel and $H(Z)$ the entropy of the source.*

The function f in the above theorem combines the source encoding with choosing a code for the channel. Notice that it is transmitting $\frac{l}{k}$ symbols per channel use. The theorem states that it is impossible to transmit more than $\frac{C}{H(Z)}$ symbols per channel use with arbitrarily low probability of error, same as if we treated the two problems separately.

10. Summary

We have introduced entropy for a random variable X with finitely many possible values and interpreted it in various ways. Initially entropy was defined as the expectation of the self-information of the simple events $\{x_i\}$, which in a certain setting was the number of bits of information needed to describe $\{x_i\}$. Since $0 \leq H(X) \leq \log N$, with the bounds attained by the degenerate and uniform distribution, respectively, the entropy can be thought of as the randomness of X. In the setting of the asymptotic equipartition property, we saw that the distribution of $(X_1, \ldots, X_n)$ was approximately a uniform distribution on a set $A_\epsilon^{(n)}$, the size of which was $2^{nH(X)}$. Finally, in the setting of source coding, $H(X)$ was the rate of information generation of the source X, measured in bits per symbol. Viewed in another way, it was the logarithm of the "effective" number of possible values of X.

The conditional entropy $H(X|Y)$ is the remaining entropy of X, after observing Y. The reduction in the entropy of X as a result of observing Y is the mutual information of X and Y. We saw that simply transforming Y cannot increase

the mutual information, but it can preserve it, if the transformation is a sufficient statistic of Y for the parameter X. Next we discussed the channel coding problem and we saw that the mutual information between the input and output of the channel is what determines its capacity for (almost) error-free transmission. Finally, we saw that the problems of source and channel coding can be treated separately, without loss of efficiency relative to the combined treatment.

References

1. T. Cover, J. Thomas *Elements of Information Theory*, 2nd Ed, John Wiley & Sons, Inc., 2006.
2. O. Kallenberg *Foundations of Modern Probability*, Springer, 2ed.
3. C. Shannon *A Mathematical Theory of Communication*, The Bell System Technical Journal, Vol 27, 1948.