Lecture Notes for Communication Theory

February 15, 2011

Please let me know of any errors, typos, or poorly expressed arguments. And please do not reproduce or distribute this document outside Oxford University.
David Stirzaker.

Contents

1 Introduction

This section outlines the basic problems that communication theory has been developed to solve, introduces the main ideas that are used to deal with those problems, and sketches their solutions. In other words, it amounts to a preview and synopsis of the rest of the course.

1.1 Basics

The principal problem of communication is to accept a message at some point, and then reproduce it as efficiently, reliably, and securely as possible at some other point. The first step in this is taken by the engineers who make a suitable mechanism M for transferring the message, in the form of a **signal** of some kind; this mechanism will be called the **channel** whatever its actual physical form, which may be a wire, fibre, disc, aerial and receiver, tape, book and so on. The two ends of the channel are called the **source** and the **receiver** (which may be the same if a message is stored for retrieval) and may be separated in time or space, or both.
Communication theory begins at this stage, by supposing that there exists such a channel that is imperfect in the senses of being finite, and noisy or insecure, or all three of these. If the channel were capable of transmitting an unbounded number of symbols, arbitrarily quickly, with any desired degree of accuracy, and in total privacy, then no further work would be needed; but no such channels exist. So, as mentioned above, the theorist has three tasks:

1. We seek to transmit the message efficiently, by which we mean that the signal should make the minimum use of the channel required to convey the message to the receiver. Typically, channels cost money to use, or have competing demands for their time, or both.

2. We want the message to be sent reliably, by which we mean that ideally the receiver gets exactly the original message, with no errors. More realistically, we may ask only that the message arrives with an arbitrarily small chance of an error.

3. We may often wish our message to be private, that is to say secret, from others. By this we mean that a spy who records the signals passing through the channel will be no nearer to knowing what the original message is. Sometimes it is sufficient to require only that working out what the original message was (though possible for the enemy) would need an impractical amount of effort. Other aspects of privacy that might be desired include the receiver being confident that the enemy cannot alter the message, (or substitute an entirely fresh one), and being able to prove the authenticity and integrity of the received signal to somebody else.

These are strong requirements, but a remarkable sequence of ideas initiated by Claude Shannon, and much developed since, has shown that they are achievable; at least up to a certain well-defined level in each case. The key idea and technique which makes all this possible is that of coding, which we now consider.

1.2 Coding

Broadly, to encode something (such as a message) is to replace it by something else; this will usually be a sequence of symbols, in the context of communication theory. More formally, we can define a code (or code system) as a mapping (or function) from the set of all possible messages to a suitable set of strings of symbols. Together with another map from strings of symbols to possible messages, which we call decoding. The message and the encoding need not (and usually are not) drawn from the same collection of symbols or objects (which we generally call an alphabet). We are all familiar with specific examples of coding.

[a] In coding for secrecy, called cryptography, the intention is to replace the message by a different signal, or encryption, in such a way that the recipient can decrypt it to recover the original message; but, at the same time, an enemy reading the signal will find it unintelligible and not decodable. Classic elementary methods include simple substitution codes (replace A by B, B by C, etc), more complex substitutions such as the Playfair code, and extremely complicated systems such as the Enigma code. Encrypting the enormous amount of confidential messages that computers send to each other requires much cleverer cryptosystems, with a mathematical theory to develop them. [Which is **not** on our syllabus.]

[b] In coding for efficiency, one seeks to minimize the cost of using the channel. Usually, for obvious reasons, this is done by encoding the message to make the signal transmitted as short as possible. Here the classic example is that of Morse code which uses three symbols dot, dash and space, which may be represented as $\cdot, -$ and s. Then in encoding the roman alphabet the code assigns: $E \to \cdot s$, $T \to -s$, $Q \to --\cdot - s$, $Z \to --\cdot\cdot s$, and so on. The point here is that in many languages using the roman alphabet, E and T are common letters, whereas Q and Z are not so common. The overall length of the transmission should therefore be shorter than if equal-length codewords had been used. Likewise naval signal flags have a flag for each letter and digit, but also single flags to represent frequently occurring messages. Road signs use the same principle, with simple ideograms for common instructions, while spelling out unusual orders. Coding for efficiency is often called source coding, or data compression.

[c] In coding for reliability, (often called channel coding) we seek to ensure that the receiver can still reconstruct the original message even when the channel corrupts the signal by making random errors of transmission. This is called noise, and results in symbols being wrongly sent as a different symbol, or simply not sent at all. A very simple method of coding to mitigate the problem is the repetition code, in which each symbol is repeated a given number (such as 3, say) of times, and the most frequent symbol in this block is assumed by the receiver to have been the one sent. (This is clearly not very efficient.) A more subtle form of coding to counteract errors is the familiar checksum. It is characteristic of all such error-correcting codes that they introduce extra symbols (redundancy) to counteract the noise.

With these examples in mind, these formal definitions are natural. It is assumed that a source S is supplying a sequence of symbols (which we may call the message) from an

alphabet A. A message of length n is denoted by $\mathbf{x} \in A^n$, where A^n is the set of all strings of length n of symbols from A. The set of all finite strings of symbols from A is denoted by $A^*\cdot$.

Definition: a code $c(\cdot)$, (or encoding, or code function), for the source S is a mapping from A^* to the set B^* of finite-length strings from an alphabet B, which may be called codewords. [If the mapping is to B^m, for some m, then the code is said to be a block code.] Formally

$$c(\cdot)\colon \mathbf{x} \in A^* \to c(\mathbf{x}) \in B^*.$$

In addition, there is a decoder $d(\cdot)$ which maps B^* to the set of possible messages.

The length of the codeword $c(\mathbf{x})$ is denoted by $|c(\mathbf{x})|$. For efficiency we would like $|c(\mathbf{x})|$ to be small in the long run; for reliability we would like $d(c(\mathbf{x})) = \mathbf{x}$ as often as possible; for secrecy we wish any enemy who knows $c(\mathbf{x})$ not to be able to identify $\mathbf{x}$ in general. Briefly, the core of communication theory is devising good codes.

Among the various properties that good codes might have, this one is clearly almost essential.

Definition: a code $c(\cdot)$ is uniquely decipherable if the concatenation $c(x_1)c(x_2)\dots c(x_n)$ of codewords of symbols (or messages) from S is the image of (corresponds to) at most one sequence $x_1 \dots x_n$.

An important class of uniquely decipherable codes is this:

Definition: a code is a prefix (or instantaneous) code if no codeword is the prefix of another. (Which is to say that we cannot add letters after some $c(x)$ to get another codeword $c(y) = c(x)b_1 \dots b_m$.]

Example. Telephone numbers are a prefix code.

The mathematical codes defined above are crucial in communication theory, but the broader concept of coding is of much wider application. For example, the sequence of amino acids in DNA encodes a number of physical attributes of the individual in question. For more wide-ranging applications we note that musical notation encodes the music itself. Maps encode various features of the surface of the earth. Plans and elevations encode buildings. After some thought, you will realise that speech encodes your thoughts, and writing encodes your speech. This in turn can be given in Morse code. The ultimate conclusion of this process is a **binary encoding**, which is a string of symbols using an alphabet of just two symbols $[0, 1]$. After a little more thought you may agree that anything of practical interest in communication must be capable of encoding as a string of symbols. [You may care to recall Wittgenstein's remark: "whereof we cannot speak, thereof one must be silent".]

1.3 Source and channel

The message to be communicated is supplied by the source, about whose nature we need not be specific, but it has three key properties. First, by what we have said above, we can assume that the message comprises a finite string of symbols. [For if it were not, we would simply encode it as such.] Secondly, the message is to be selected from a set of possible messages; (which we shall assume to be finite, for simplicity). And thirdly we are uncertain about what the message is to be, because if it were known in advance, it would be unnecessary to send it. It is therefore natural to regard the output of a source as a

random sequence of symbols, which we refer to as random variables and vectors (with a slight abuse of the convention that these shall be real-valued).

Definition: a discrete source comprises a sequence of random variables $X_1, X_2, \ldots$ taking values in a finite alphabet A. Any finite string is a **message**. If the X_r are independent and identically distributed, then the source is said to be **memoryless**, and we can write $P(X_r = x) = p(x)$ for all r.

At this stage, we shall assume that sources are discrete and memoryless. [Of course, many real sources do not have these properties, but the ideas and methods that we shall develop in this simple case can be generally extended to deal with more complicated sources.] Thus the probability that the source emits a message $\mathbf{x} = (x_1, \ldots, x_n) \in A^n$ is

$$P(\mathbf{X} = \mathbf{x}) = P(X_1 = x_1, \ldots, X_n = x_n) = P(X_1 = x_1) \ldots P(X_n = x_n) = \prod_{r=1}^{n} p(x_r)$$

by the independence.

This is encoded as a signal to enter a channel:

Definition: given an alphabet B of possible input symbols, and an output D of possible output symbols, a discrete channel is a family of conditional distributions $p(y|x),\ x \in B,\ y \in D$. This array is called the channel matrix and denoted by M, so that for input X and output Y

$$M = P(Y = y|X = x) = p(y|x)$$

Since $\sum_y p(y|x) = 1$, M is a stochastic matrix.

It may be square, and it may be doubly stochastic, (i.e., $\sum_x p(y|x) = 1$), but not usually. More generally the rth extension of the channel is the family of conditional joint distributions of r uses of M, given the input $(x_1, \ldots, x_r) = \mathbf{x}$

$$p(y_1, \ldots, y_r|x_1, \ldots, x_r) = P(\mathbf{Y} = \mathbf{y}|\mathbf{X} = \mathbf{x})$$

The channel is said to be memoryless, and denoted by DMC, if

$$p(y_1, \ldots, y_r|x_1, \ldots, x_r) = \prod_{i=1}^{r} p(y_i|x_i)$$

Thus uses of the channel are conditionally independent, given the input. We shall always assume this.

We note two extreme cases:

(a) If $\mathbf{Y} = \mathbf{X}$, so that $p(y|x) = 1, whenever\ x = y \in B$, (and $p(y|x)$ is 0 otherwise) then the channel is **perfect**, or **noiseless**.

(b) If $p(y|x)$ does not depend on x, (i.e. $p(y|x) = p(y)$ for all x), then the output is pure noise, independent of the input, and the channel is **useless**.

1.4 Entropy

Given a source, we do not know in advance what the message is to be, but we can ask how likely it is that we will see any given sequence $\mathbf{x} = (x_1, \ldots, x_n)$. The likelihood of $\mathbf{x}$ is simply the probability that S emits $\mathbf{x}$, namely (by the independence)

$$p(\mathbf{x}) = \prod_{r=1}^{n} p(x_r)$$

Before the source produces it, $\mathbf{x}$ may take any of its possible values, so the likelihood of the actual message that we are to see is a random variable called the empirical likelihood

$$p(\mathbf{X}) = \prod_{r=1}^{n} p(X_r)$$

It often turns out that for various purposes sums are more tractable than products, so the following is natural.
Definition: the empirical log-likelihood (function) of the source is

$$L(\mathbf{X}) = -\sum_{r=1}^{n} \log p(X_r)$$

where by convention logarithms in communication theory are to base 2, unless otherwise stated. [We shall see why this is later.] The application of the negative sign is also just a convenient convention, to make $L(\mathbf{X})$ non-negative. Since X is a simple random variable, $L(X) = -\log p(X)$ has an expectation.
Definition: the expected value of $L(X)$ is denoted by

$$H(X) = -E \log p(X) = -\sum_{x \in A} P(X = x) \log P(X = x),$$

and called the entropy of X; (and also the information in X, or the uncertainty in X, for reasons that we discuss later). Perhaps surprisingly, this function is of great importance in communication theory, as we shall see.
Notes:

(1) By convention, we take $0 \log 0 = 0$, when $p(x) = 0$.

(2) $H(X)$ does not depend on the values of X, only on its distribution $(p_1, \ldots, p_n)$. We sometimes use the notation $H(p_1, \ldots, p_n) = H(\mathbf{p}) = H(X)$.

(3) The entropy of any random vector (X, Y), or $\mathbf{X}$, is defined in exactly the same way by the joint distribution.

Example. Let X be Bernoulli(p). Then

$$H(X) = H(p, 1-p) = H(\mathbf{p}) = -p \log p - (1-p) \log(1-p)$$

Sketch this curve as a function of p, and note that it is maximal for $p = \frac{1}{2}$, when $H(\frac{1}{2}, \frac{1}{2}) = 1$. This corresponds to the flip of a fair coin, and this determines the units of

entropy called bits. Thus the flip of a fair coin has unit entropy, because we chose to take logarithms to base 2.

Lemma.

$$H(\mathbf{X}) = \sum_{r=1}^{n} H(X_r) = nH(X)$$

for the discrete memoryless source.

Proof.

$$H(\mathbf{X}) = -E \log p(\mathbf{X}) = E \log \prod_{r=1}^{n} P(X_r) = -E \sum_{r=1}^{n} \log p(X_r) = \sum_{r=1}^{n} H(X_r)$$

Likewise it is shown that if X and Y are independent then

$$H(X, Y) = H(X) + H(Y)$$

Lemma: $H(X) = 0$ if and only if X is a constant with probability 1.

Proof. Each term in the sum is zero iff either $p_X(x) = 0$ or $p_X(x) = 1$. There must thus be just one x with the second property.

Lemma. Let $c(\cdot)$ be an invertible (uniquely decipherable) encoding of $\mathbf{X}$. Then the entropy of $c(\mathbf{X})$ is the same as that of $\mathbf{X}$. We interpret this as the important result: an invertible code neither increases uncertainty nor loses information.

Proof. Let $\mathbf{Y} = c(\mathbf{X})$. Then (in an obvious notation)

$$H(\mathbf{Y}) = -\sum p_Y(y) \log p_Y(y) = -\sum P(c(\mathbf{X}) = y) \log P(c(\mathbf{X}) = \mathbf{y})$$

$$= -\sum P(\mathbf{X} = c^{-1}(\mathbf{y})) \log P(\mathbf{X} = c^{-1}(\mathbf{y}))$$

$$= H(\mathbf{X})$$

, using the unique decipherability.

We turn from these useful lemmas to an important result, which we shall use often;

Theorem. **Gibbs inequality**. Let X have distribution $p(x)$, and let $q(x),\ x \in A$, be any other probability distribution on the same alphabet as $p(x)$. Then $H(X)$ uniquely minimizes the value of the function $G(q) = -E \log q(X)$ over all choices of q. That is to say, for any distributions p and q on A

$$H(X) = -\sum p(x) \log p(x) \leq -\sum p(x) \log q(x)$$

with equality if and only if $p(x) = q(x),\ x \in A$.

Proof of Gibbs' inequality.
We give two proofs.
For the first, recall Jensen's inequality for a strictly convex function, $u(X)$ of a random variable X; that is: $Eu(X) \geq u(EX)$ with equality iff X is constant, so that $X = EX$. Now $u(x) = -\log x$ is strictly convex for $x > 0$. Therefore, letting X have distribution $p(x)$,

$$\sum p(x) \log \frac{p(x)}{q(x)} = E\left\{-\log \frac{q(X)}{p(X)}\right\} \geq -\log\left(E\frac{q(X)}{p(X)}\right) = -\log\left[\sum_x \frac{q(x)}{p(x)} p(x)\right] = 0,$$

with equality iff

$$\frac{q(x)}{p(x)} = \text{ constant } = E\frac{q(X)}{p(X)} = 1$$

, so that $p(x) = q(x)$ for all x.
For the second proof, recall that $\log_b x \leq \dfrac{x-1}{\log_e b}$, for $b > 1$ and $x > 0$, with equality iff $x = 1$. Hence

$$-\log_e 2 \sum_x p(x) \log \frac{p(x)}{q(x)} = \sum_{p>0} p(x) \log_e \frac{q(x)}{p(x)}$$

$$\leq \sum_{p>0} p(x)\left\{\frac{q(x)}{p(x)} - 1\right\} \text{ with equality iff } p(x) = q(x) \text{ for all } p(x) > 0$$

$$= \sum_{p>0} q(x) - 1$$

$$\leq 0, \text{ with equality iff } \sum_{p>0} q(x) = 1,$$

which entails $q(x) = p(x)$ when $p(x) = 0$. Hence equality holds throughout iff $p(x) = q(x)$ for all x.
Here are some useful consequences.
Corollary. $H(X) \leq \log |A| = \log a$, with equality if and only if X is uniformly distributed on A.
Proof. Let $q(x) = |A|^{-1} = a^{-1},\ x \in A$. Then

$$H(X) \leq -\sum_{x \in A} p(x) \log \frac{1}{a} = \log a,$$

with equality if and only if $p(x) = a^{-1}$.

We interpret this by regarding $H(X)$ as a measure of how "spread out" the distribution of X is over its possible letters.
Now, rearranging Gibbs inequality, we find that if we regard it as a function of the two distributions we have this

$$\sum_x p(x) \log \frac{p(x)}{q(x)} :\ = d(p, q) \geq 0,$$

with equality iff $p \equiv q$. The function $d(p,q)$ defined by the above sum can thus be seen as a measure of how far the two distributions differ from each other. [We cannot see it as a strict distance, because $d(p,q) \neq d(q,p)$.]
Note that in the defining sum we adopt the conventions that for $p \neq 0$, $p \log \frac{p}{0} = \infty$, whereas $0 \log \frac{0}{q} = 0$ for any value of q. Thus we have :
Definition. The function $d(p,q)$ is the relative entropy between the distributions $p(x)$ and $q(x)$, $x \in A$; it is also called the Kullback-Leibler divergence. [It may be referred to either as a similarity measure, or a dissimilarity measure, depending on your point of view.]
And some writers call it the information-theoretic divergence.
Example. Let X with distribution $p(x) = \frac{1}{2}$ be Bernoulli($\frac{1}{2}$); and Y be Bernoulli(r), with distribution $q(1) = P(Y = 1) = r$, and $q(0) = P(Y = 0) = s = 1 - r$.

Then an easy calculation gives

$$d(p,q) = -1 - \frac{1}{2}\log r - \frac{1}{2}\log(1-r)$$

$$= 0 \text{ iff } r = \frac{1}{2}$$

, and likewise

$$d(q,p) = 1 + r\log r + (1-r)\log(1-r)$$

$$= 0 \text{ iff } r = \frac{1}{2}$$

.

These are the relative entropies between a fair coin and a biased coin, depending on which is taken first.
Finally, we note the most important property of all: as n increases the average empirical log-likelihood of $\mathbf{X} = (x_1, \ldots, x_n)$ per source symbol converges (in probability) to the entropy $H(X)$.
Theorem. For $\delta > 0$, as $n \to \infty$,

$$(*) \qquad P(|\frac{1}{n}\log p(\mathbf{X}) + H(X)| > \delta) \to 0$$

Proof. First we recall Chebyshov's inequality:

$$P(|X| > \delta) \leq E|X|^2/\delta^2 \quad \text{for } \delta > 0$$

. [To see this note that $\delta I(|X| > \delta) \leq |X|$, where $I(A)$ is the indicator of the event A, so if we square this and take the expected value

$$\delta^2 EI^2 = \delta^2 P(|X| > \delta) \leq E|X|^2].$$

Hence the probability in (*) is less than or equal to

$$\frac{1}{\delta^2 n^2} E[\log p(X) + nH(X)]^2 = \frac{1}{\delta^2}\frac{1}{n^2} \text{ var } L_n = \frac{1}{n\delta^2} \text{ var } L_1 \to 0$$

as $n \to \infty$ since L_1 has finite variance. [Note that this is essentially a simple weak law of large numbers.]

We use this key theorem in the next section to show that although the total number of possible messages of length n is $|A|^n$, in fact, with probability arbitrarily close to 1, $\mathbf{X}$ is a message lying in a set T_n of messages that is much smaller than A^n, except when X is uniform on A.

1.5 Typicality

We have shown above that the entropy of a message $\mathbf{X}$ of length n from the source is $nH(X)$, where $H(X)$ is the entropy of any letter. Before the message appears, not much can be said about any particular symbol, except its distribution $p(x)$. But suppose we consider arbitrarily long messages from the source. Claude Shannon's remarkable insight was that such messages have this property.

Theorem. Typicality.

Consider a discrete memoryless source. Then for $\epsilon > 0$ and $\delta > 0$ there exists $n_0 < \infty$ such that for all $n > n_0$ the set A^n of all possible sequences of length n can be divided into disjoint sets T_n and U_n such that $T_n \cup U_n = A^n$ and

(1) $2^{-n(H+\delta)} \le p(\mathbf{x}) \le 2^{-n(H-\delta)}$, for $\mathbf{x} \in T_n$,

(2) $P(\mathbf{X} \in T_n) \ge 1 - \epsilon$

(3) $(1-\epsilon)2^{n(H-\delta)} \le |T_n| \le 2^{n(H+\delta)}$

That is to say, more informally, as n increases A^n can be split into a set U_n of arbitrarily small probability (called the untypical sequences) and a set T_n of probability arbitrarily near 1, by (2), called the typical set. Thus for many practical purposes we can treat the messages of length $n > n_0$ as though there were only 2^{nH} of them, by (2) and (3), and with each such typical message having roughly the same probability 2^{-nH} of occurring, by (1).

The point of this is that from above, for some $\gamma > 0, H(X) \le \log|A| - \gamma$, provided that X is not uniform on A. Hence, choosing $\delta < \gamma, |T_n| \le 2^{n(H+\delta)} \le 2^{n(\delta-\gamma)}|A|^n$ and we see that the set of typical messages is much smaller than the set of possible messages in the long run. This idea makes possible both Shannon's source and channel coding theorems, as we see in the following sections. [Note the slightly counter-intuitive fact that the most probable messages are not typical.]

Proof of the theorem.

Define the typical set T_n to be those messages $\mathbf{x}$ whose log-likelihood is within a distance δ from H. That is to say

$$T_n = \{\mathbf{x} : \left|\frac{1}{n}\log p(\mathbf{x}) + H(X)\right| < \delta\}$$

Rearranging the inequality gives (1). Now using the empirical log-likelihood convergence theorem of the previous section gives (2). It follows that

$$1 - \epsilon \leq P(\mathbf{X} \in T_n) = \sum_{\mathbf{x} \in T_n} p(\mathbf{x}) \leq 1,$$

and now applying the two bounds in (1) to each $p(\mathbf{x})$ in the sum gives (3). For example,

$$1 \geq \sum_{\mathbf{x} \in T_n} p(\mathbf{x}) \geq \sum_{\mathbf{x} \in T_n} 2^{-n(H+\delta)}$$

$$= |T_n| 2^{-n(H+\delta)}$$

, so that $|T_n| \leq 2^{n(H+\delta)}$.
This theorem is sometimes called the Asymptotic Equipartition Property, or AEP.
Finally, we note that the idea of typicality can be formulated more strongly. The results above address only the probability of a sequence $\mathbf{x}$, so that a sequence $\mathbf{x}$ is typical if $|\frac{1}{n} \log p(\mathbf{x}) + H| < \delta$. This tells us little about the actual sequence itself, that is to say the actual frequency of occurrence of the letters of A in the message $\mathbf{x}$. Strong typicality addresses exactly that; so we define $N(\alpha, \mathbf{x})$ to be the number of occurrences of $\alpha \in A$ in $\mathbf{x}$. The collection $[N(\alpha, \mathbf{x}) : \alpha \in A]$ is called the type of $\mathbf{x}$.
Definition. Let $\delta > 0$. The message $\mathbf{x} \in A^n$ is said to be δ-strongly typical for $p_X(x)$ if

$$\left| \frac{1}{n} N(\alpha, \mathbf{x}) - p_X(\alpha) \right| < \delta \text{ when } p_X(\alpha) > 0$$

, and $N(\alpha, \mathbf{x}) = 0$ whenever $p_X(\alpha) = 0$.
That is to say, the empirical distribution $N(\alpha, \mathbf{x})$ is close to the source distribution $p_X(\alpha)$; (in total variation distance, more formally). The set of such sequences is called the strongly typical set, and it turns out to have essentially the same properties as the weakly (or entropy) typical set. That is to say, its probability is arbitrarily close to 1, and its sequences are asymptotically equiprobable. This may be called the strong AEP.

1.6 Shannon's first theorem: noiseless (or source) coding

Recall that our task is to use the channel efficiently; an obvious way to do this is to seek a code that minimizes the expected length of the encoded message, or signal, passing through the channel. Remarkably, Shannon showed this:
Theorem. If a source having entropy $H(X)$, is encoded using an alphabet B, of size $b = |B|$, then given $\epsilon > 0$, for large enough n there is an encoding function $c(\cdot)$, from A^n to $B^m \cup B^k$, for some $k,\ m \geq 1$, such that

$$\frac{1}{n} E|c(\mathbf{X})| \leq \frac{H(X)}{\log b} + \epsilon$$

That is to say, the expected number of signal symbols per symbol of $\mathbf{X} = (X_1, \ldots, X_n)$ is arbitrarily close to $\dfrac{H(X)}{\log b}$, as $n \to \infty$.
Conversely, no such invertible block encoding using B can have shorter expected length

than this in the long run. Note that the result is particularly neat for binary encodings when $b = 2$.

Proof. By the typicality theorem, for large enough n there exists a set T_n such that $P(\mathbf{X} \in T_n) \geq 1 - \epsilon$ and $|T_n| \leq 2^{n(H+\delta)}$.

Choose m to be the smallest integer such that $|B|^m \geq 2^{n(H+\delta)}$, so that $m \leq \dfrac{n(H+\delta)}{\log b} + 1$.

Now construct an encoding (codebook) as follows. Because there are more m-strings than typical messages of length n, each element of T_n can be invertibly encoded by a distinct m-string prefixed by $b_0 \in B$. Then the untypical set can be invertibly encoded using k-strings from B^k, for any k such that $|B|^k \geq |A|^n$, prefixed by $b_1 \in B$; $b_1 \neq b_0$.

Then

$$\frac{1}{n}E|c(\mathbf{X})| = \frac{1}{n}\left\{\sum_{\mathbf{x}\in T_n} p(\mathbf{x})(m+1) + \sum_{\mathbf{x}\in U_n} p(x)(k+1)\right\}$$

$$\leq \frac{m+1}{n} + \frac{k+1}{n}\epsilon$$

$$\leq \frac{H+\delta}{\log b} + \frac{2}{n} + \frac{k+1}{n}\epsilon.$$

Since δ and ϵ, and then n, are chosen arbitrarily, the first result follows.

Conversely, we can use the AEP (typicality) to show that no binary block code can have block length less than $nH(X)$. To see this consider the sequence $(X_1, \ldots, X_n)$, and a possible invertible binary encoding in blocks of length m, for some $m, (Y_1, \ldots, Y_m)$. By the lemma in 1.4, because $(Y_1, \ldots, Y_m)$ is an invertible function of $(X_1, \ldots, X_n)$ it has the same entropy, namely $nH(X)$. By the AEP, there are asymptotically $2^{nH(X)}$ messages from the source as n increases. By the note in 1.4, the maximum entropy of $(Y_1, \ldots, Y_m)$ is $\log 2^m = m$ in which case there are 2^m typical strings in the encoding. For invertibility, $2^m \geq 2^{nH(X)}$, as required.

The same result holds for invertible encodings of variable length; we prove this later on. Thus Shannon's entropy $H(X)$ provides the explicit universal lower bound for the extent to which messages may be compressed for efficient transmission.

1.7 Information

As remarked above, channels are not perfect and signals are often corrupted. That is to say, if the correct signal received should be Y, the actual signal received may be some function $u(Y)$ of Y. If this is a non-random function, the effect is called distortion; in this case the originally intended result Y may be recovered if $u(\cdot)$ is invertible, and its form discovered by trial messages. More commonly $u(\cdot)$ is random, and the effect is called noise. Our canonical definition of a noisy channel is this:

Definition. A discrete channel comprises a family of conditional distributions, $p(y \mid x) = p(Y = y \mid X = x)$ where $X \in A$ is the input and $Y \in B$ is the output. It is said to be memoryless if outputs depend only on their corresponding input, and are conditionally independent given the input message $\mathbf{X}$. That is

$$p(\mathbf{y} \mid \mathbf{x}) = p(\mathbf{Y} = \mathbf{y} \mid \mathbf{X} = \mathbf{x}) = p(y_1 \mid x_1)p(y_2 \mid x_2)\ldots p(y_n \mid x_n)$$

These can be seen as the conditional probabilities defining a channel with input alphabet A^r, and output alphabet B^r, and this may be called the rth extension of the channel $p(y \mid x)$. The array $p(y \mid x),\ x \in A,\ y \in B$ is called the channel matrix. It is stochastic, and may or may not be a square matrix.
For any given input distribution $p_X(x)$, the input and output have joint distribution

$$p(x, y) = p_X(x)p(y \mid x)$$

Thus X and Y have respective entropies $H(X)$ and $H(Y)$, and joint entropy $H(X, Y)$. However, in the context of a noisy channel it is natural to consider yet another entropy function: the entropy of the distribution $p(y \mid x)$ for any fixed x. This is given by

$$H(Y \mid X = x) = -\sum_y p(y \mid x) \log p(y \mid x)$$

and called the conditional entropy of Y given $X = x$. Note that as x ranges over A, this defines a random variable, being a function of X. It therefore has an expectation, which is the expected value of the entropy in Y, conditional on the value of X, before the input symbol is supplied. It is given by

$$\begin{aligned} H(Y \mid X) &= \sum_x p_X(x) H(Y \mid X = x) \\ &= -\sum_{x,y} p(x, y) \log p(y \mid x) \\ &= -E \log p(Y \mid X) \end{aligned}$$

[This is not a random variable of course, despite the similarity of notation with conditional expectation $E(Y \mid X)$ which is a random variable.]
Lemma $H(X \mid Y) \geq 0$, with equality iff X is a non-random function of Y; ie $X = g(Y)$ for some $g(\cdot)$.
Proof. The non-negativity is obvious. Now $H(X \mid Y) = 0$ iff $H(X \mid Y = y) = 0$ for all y. But any entropy is 0 iff the distribution is concentrated at a point, so that $x = g(y)$ for some g and all y.
We return to this entropy later, but note that $H(X \mid Y)$ is of particular interest, as it represents the expected uncertainty of the receiver of the transmitted signal about what was actually sent. It has been called the equivocation.[And $H(Y \mid X)$ has been called the prevarication.]
Now recall that we mentioned two extreme cases, useless channels in which the output is noise independent of the input, and perfect channels with no noise. Obviously in intermediate cases it would be very useful to have some measure of just how good (or bad) the channel is; that is to say, how close Y is to X, in some suitable sense. We would then (we hope) be able to choose $p_X(x)$ to make Y as close to X as possible, thus optimizing the channel's performance. Fortunately, we have already defined such a measure of closeness above, in the form of the relative entropy (or Kullback-Leibler divergence). We therefore judge our channel by how far it is from being useless, namely the relative entropy between $p(x, y)$ and $p_X(x)p_Y(y)$.
Definition. For random variables X and Y, (seen as the input signal and output signal

respectively), their mutual information $I(X;Y)$ is the relative entropy between $p(x,y)$ and $p_X(x)p_Y(y)$

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p_X(x)p_Y(y)} = E \log \frac{p(X,Y)}{p_X(X)p_Y(Y)}$$

When X and Y are independent $I(X;Y) = 0$, and the channel is useless. When $X = Y$, $I(X;Y) = H(X)$, and the input and output have the same entropy, as expected. In intermediate cases we may choose $p_X(x)$ to get the best we can from the channel. This definition is therefore natural:

Definition. The (Shannon) capacity of a channel with input X and output Y is

$$C = \max_{p_X(x)} I(X;Y)$$

As with $H(X)$, this definition will be further justified by its applications, to follow. For the moment, we note some properties of $I(X;Y)$, and its relationship to entropies.

Theorem.

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

with equality in the last line if and only if X and Y are independent.

Proof. All follow from the definitions, except the last assertion which is a consequence of Gibbs inequality, yielding equality when $p(x,y) = p_X(x)p_Y(y)$, as required.

Corollaries

1. $H(X) \geq H(X \mid Y)$ with equality iff X and Y are independent, (informally we recall this as **conditioning reduces entropy**)
2. $H(X,Y) \leq H(X) + H(Y)$ with equality iff X and Y are independent
3. $H(X,Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y)$
   This is called the **chain rule**.
4. For non-random $g(\cdot)$
   (a) $H(g(X)) \leq H(X)$, with equality iff g is invertible
   (b) $H(X, g(X)) = H(X)$

Proof. 1-3 are trivial. For 4, recall that $H(g(X) \mid X) = 0$, so we obtain

$$(b) \quad H(X, g(X)) = H(X) + H(g(X) \mid X) = H(X)$$

$$(a) \quad H(X, g(X)) = H(g(X)) + H(X \mid g(X)) \geq H(g(X))$$

with equality iff $H(X \mid g(X)) = 0$,
which means X is a function of $g(X)$, as required for the invertibility.

Example. Shannon noiseless coding bound.
Let a source sequence $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ be encoded as binary strings by a uniquely

decipherable function $c(\cdot)$ yielding the binary signal $\mathbf{Y} = c(\mathbf{X}) \in \{0,1\}^*$. Then denoting the length of $c(x)$ by $|c(x)|$, $E|c(X_1)| \geq H(X_1)$

Proof. Let $M = c(X_1)\dots c(X_n)$ be the concatenation of length L. Since M is uniquely decodable, (invertible), $H(M) = nH(X_1)$, by the memorylessness of the source.
Also, with probability 1, $L \leq n \max_{x\in A}|c(x)|$, so $H(L) \leq \log(n \max_x |c(x)|)$
Finally, since L is a function of $\mathbf{X}$ and of M,

$$nH(X) = H(M) = H(M,L) = H(L) + H(M \mid L) \quad \text{, by the chain rule,}$$

$$= H(L) + \Sigma H(M \mid L = k)P(L = k)$$

$$\leq \log(n \max_x |c(x)|) + \sum_k \log 2^k P(L = k)$$

$$= \log(n \max_x |c(x)|) + EL = \log(n \max_x |c(x)|) + nE|c(X)|$$

Dividing by n, and letting $n \to \infty$ yields the result.
Finally in this section, we note that just as the entropy of X may be defined conditional on some random variable Y, so too may the mutual information of X and Y be defined conditional on some random variable Z. [Provided, as always, that they are all jointly distributed.]
Definition. The conditional mutual entropy of X and Y given Z, i.e., the conditional mutual information is

$$I(X;Y \mid Z) = d(p(x,y \mid z), p(x \mid z)p(y \mid z)) \geq 0$$

with equality iff X and Y are conditionally independent given Z, which is to say that

$$p(x,y \mid z) = p(x \mid z)p(y \mid z)$$

We also have this useful result:
Lemma. Chain rule for information

$$I(X,Y;Z) = I(X;Z) + I(Y;Z \mid X)$$

Proof. An easy exercise for you.

1.8 Shannon's second theorem. Noisy (or channel) coding.

The typicality theorem (or AEP) used to demonstrate the noiseless coding theorem can also be used to establish a bound on the rate of reliable transmission that can be achieved using a noisy channel. Formally:
Theorem. Suppose that a source, of entropy H, produces a message of arbitrary length n, which it is desired to encode for transmission through a channel of capacity C. If $H \leq C$, it is possible to encode a sufficiently long message as a signal in such a way that the received (output) signal can be decoded with arbitrarily small probability of error. If $H > C$, then this is not possible, in the sense that the probability of error can not be made arbitrarily small.
We give what is essentially Shannon's argument.

Sketch proof. Let X and Y be the input and output of the channel with capacity C, and let X have the distribution $p_X(x)$ that actually achieves the capacity C. [In practical cases, $I(X;Y)$ is a continuous function on a closed bounded subset of $\mathbb{R}^a$, so the supremum over $p_X(x)$ is indeed attained.] For arbitrarily large n, consider the sequence of inputs $(X_1, \ldots, X_n)$, where these are i.i.d. with distribution $p_X(x)$.
By typicality, (the AEP), we have that

1. There are about $2^{nH(X)}$ typical inputs $\mathbf{x}$.
2. There are about $2^{nH(Y)}$ typical outputs $\mathbf{y}$.
3. The conditional entropy of the input X given the output Y is $H(X \mid Y)$, and thus, also by typicality, to each typical output there corresponds on average about $2^{nH(X|Y)}$ typical inputs.

To see this another way, consider the input and output together as a single random vector with entropy $H(X,Y)$. Shannon's theorem (the AEP) shows that there are about $2^{nH(X,Y)}$ input-output pairs, which are often called jointly typical sequences. Thus there are about

$$\frac{2^{nH(X,Y)}}{2^{nH(Y)}} = 2^{nH(X|Y)}$$

typical inputs per typical outputs on average; (as seen above from the other point of view). Now suppose that there is a source producing messages with entropy rate $R_0 = H < C$; that is to say as n increases it supplies about 2^{nR_0} typical messages of length n. We wish to encode these for reliable transmission through the channel of capacity C. Choose R such that $R_0 < R < C$, and construct a coding scheme as follows:

1. From the $2^{nH(X)}$ typical input messages defined above (with distribution $p_X(x)$ achieving capacity) select 2^{nR} independently at random, with replacement; these are the codewords. [Note that since the selection is with replacement, we admit the possibility of having two codewords the same, somewhat counter-intuitively.] Denote this codebook by $\mathbf{x}(1) \ldots \mathbf{x}(2^{nR})$.
2. The decoding scheme is this: for any output $\mathbf{Y}$ we will look at the set $S(\mathbf{Y})$ of corresponding typical inputs (i.e., those that are jointly typical with $\mathbf{Y}$), of which there are typically about $2^{nH(X|Y)}$, as remarked above. If $\mathbf{x}(r)$ is sent, then with probability arbitrarily close to 1, the output $\mathbf{Y}$ will be jointly typical with $\mathbf{x}(r)$. If on examining the set $S(\mathbf{Y})$ of inputs that are jointly typical with $\mathbf{Y}$ we find no other codeword than $\mathbf{x}(r)$ then decoding $\mathbf{Y}$ as $\mathbf{x}(r)$ is correct. Otherwise, if $S(\mathbf{Y})$ contains another codeword, we declare an error in transmission.

With this codebook and decoding scheme, we pick a codeword to send at random from $\mathbf{x}(1), \ldots, \mathbf{x}(2^{nR})$. The average probability of error (averaged over random choice of codebooks and random choice of codeword to send) is therefore
$p_e = P(\text{at least one codeword not equal to that sent lies in the set } S(\mathbf{Y}))$

$$\leq \sum_{k=1}^{2^{nR}} P(\mathbf{x}(k) \in S(\mathbf{Y})), \text{ since } P(\cup A_i) \leq \sum P(A_i)$$

$$\approx \frac{2^{nR}2^{nH(X|Y)}}{2^{nH(X)}}$$

, because there are about $2^{nH(X)}$ possible choices for $\mathbf{x}(k)$, of which about $2^{nH(X|Y)}$ are in $S(\mathbf{Y})$. Hence
$p_e \leq 2^{nR}2^{-nC}$, since $p_X(x)$ achieves $C \to 0$ as $n \to \infty$, because $R < C$.
It follows that for any $\epsilon > 0$, there exists $n < \infty$ such that there is a fixed set of codewords $\mathbf{x}(1), \ldots, \mathbf{x}(2^{nR})$ that has average (over codeword selected) error smaller than ϵ. Now order these codewords by their probabilities of error, and discard the worst half (with greatest probability of error). The remaining codewords have arbitrarily small maximum probability of error, and there are $2^{n(R-\frac{1}{n})}$ codewords in the book. This exceeds 2^{nR_0} for large enough n, so the message from the source can thus be invertibly coded, with maximum probability of error as small as we choose. Note that this is purely a proof of the existence of such a codebook. There is no clue as to how we might find it, (except the essentially useless method of searching through all possible codebooks).

We conclude this section with a brief look at other popular decoding rules for noisy channels. [For noiseless channels decoding is clearly trivial, since the receiver always sees the codeword that was sent.] Formally, in general, we have this:
Definition. A decoder (or decoding function) $g(\cdot)$ is defined on all possible outputs y of the channel, and takes values in the set of all codewords, possibly augmented by a symbol e denoting that the decoder declares an error.
Example. The ideal observer (or minimum error decoder) chooses the most likely codeword given the output of the channel. Thus (in an obvious notation)

$$g(y) = \begin{cases} c(x) & \text{if there is a unique } x \text{ such that } p(c(x) \mid y) \text{ is maximal} \\ e & \text{otherwise} \end{cases}$$

This rule has a potential disadvantage, in that it is necessary to know the distribution of the codewords, $p(c)$, since

$$p(c \mid y) = p(y \mid c)\frac{p(c)}{p(y)}$$

A decoder without this problem is this:
Example. Maximum likelihood decoder.

$$g(y) = \begin{cases} c(x) & \text{if there is a unique } x \text{ such that } p(y \mid c(x)) \text{ is maximal} \\ e & \text{otherwise} \end{cases}$$

This chooses the codeword that makes the received message most likely.
Another way of defining decoders is to view the codewords and signal as points in the same suitable space, with a distance function $\|\cdot\|$.
Example. Minimum distance (or nearest neighbour) decoder.

$$g(y) = \begin{cases} c(x) & \text{if there is a unique } x \text{ such that } \|c(x) - y\| \text{ is minimal} \\ e & \text{otherwise} \end{cases}$$

When alphabets are binary, a very natural distance between binary strings $c(x)$ and y of length n is the Hamming distance, in which $\|c(x) - y\|$ is the number of places at which

$c(x)$ and y disagree. Then the Hamming decoder chooses the $c(x)$ nearest to y, if it is unique, in Hamming distance.
A variant of this is the Hamming r-sphere decoder, which chooses the $c(x)$ nearest to y, provided it is unique and differs from y in at most r places; otherwise it declares an error.

1.9 Differential entropy

In the real world, and also in statistical and engineering models, noise is often seen as normally distributed. It is thus natural to seek to define entropy for a continuous random variable X, having a density $f(x)$.
Definition. The entropy, or differential entropy, $h(X)$ of the continuous random variable X is

$$h(X) = -\int f(x) \log f(x) \mathrm{d}x = -E \log f(X)$$

where $f(x)$ is the density of X, provided that the integral exists.
Note that there are some marked differences between $h(X)$, and $H(X)$ as defined in the discrete case. First, it is customary to take logarithms to base e, (natural logarithms), for differential entropy. Second, $h(X)$ can take any value in $\mathbb{R}$. The resulting unit of information is called the nat $= \log_2 \mathrm{e}$ bits; and a bit $= \log_\mathrm{e} 2$ nats; because

$$\log_\mathrm{e} x = \log_2 x \log_\mathrm{e} 2 \text{ and } \log_2 x = \log_\mathrm{e} x \log_2 \mathrm{e}$$

Example. Let X be uniformly distributed on $(a, a+b)$, so $f(x) = \frac{1}{b}$ in this interval. Then

$$h(X) = -\int_a^{a+b} \frac{1}{b} \log \frac{1}{b} \mathrm{d}x = \log b$$

which is negative for $b < 1$.
The joint entropy $h(X, Y)$ and conditional entropy $h(X \mid Y)$ are defined analogously in the same manner as the discrete entropy. Likewise we have this:
Definition. The relative (differential) entropy between densities $f(x)$ and $g(x)$ is

$$d(f, g) = \int f(x) \log \frac{f(x)}{g(x)} = E \log \left(\frac{f(x)}{g(x)} \right)$$

Furthermore, Gibbs inequality holds:
Theorem Gibbs inequality
$d(f, g,) \geq 0$, with equality iff $f(x) = g(x)$ for all x; (except perhaps on a set of measure zero).
Proof. This can be proved in the same way as the discrete case (by either method used there), and is left as an exercise.
Hence, as in the discrete case, we have these:
Corollaries: Let X and Y have joint density $f(x, y)$, (and all the entropies are assumed to be finite), then

1. $h(X, Y) \leq h(X) + h(Y)$, with equality iff X and Y are independent
2. $h(X, Y) = h(X) + h(Y \mid X) = h(Y) + h(X \mid Y)$

3. $h(X \mid Y) \leq h(X)$, with equality iff X and Y are independent.

Furthermore, the mutual information is defined and behaves likewise.
Definition. $I(X;Y) = H(X) + H(Y) - H(X,Y) = d(f(x,y), f_X(x)f_Y(y)) \geq 0$ with equality if and only if X and Y are independent.

Example. $Q = (X,Y)$ is a random point uniformly distributed in the square determined by the four points having Cartesian coordinates $(0, \pm 1), (\pm 1, 0)$. What is the information conveyed about X by Y?
Solution. The joint and marginal densities are $f(x,y) = \frac{1}{2};\ f_X(x) = 1 - |x|;\ f_Y(y) = 1 - |y|$.
Hence,

$$I(X,Y) = E \log \frac{f(X,Y)}{f_X(X) f_Y(Y)}$$

$$= -\log 2 - E \log\{(1 - |X|)(1 - |Y|)\}$$

$$= -\log 2 - \int\int_S \log(1 - |x|)(1 - |y|) \mathrm{d}x \mathrm{d}y \text{ , by symmetry}$$

$$= -\log 2 - 4 \int_0^1 \int_0^{1-x} \log(1 - x) \mathrm{d}y \mathrm{d}x \text{ , also by symmetry}$$

$$= -\log 2 + 4 \left[\frac{1}{2}(1 - x)^2 \log(1 - x) \right]_0^1 + 2 \int_0^1 (1 - x) \mathrm{d}x$$

$$1 - \log_{\mathrm{e}} 2 \approxeq 0.31 \text{ nats } \approxeq 0.44 \text{ bits}$$

Note that if S had been the square $(\pm 1, \pm 1)$, then $I(X;Y) = 0$, as X and Y are then independent. But the covariance of X and Y is zero in both cases, so $I(X;Y)$ is a better measure of association from one point of view.
Finally, we note one important difference between $H(X)$ and $h(X)$. When X is simple and $g(X)$ is a one-one invertible function of X, we have $H(X) = H(g(X))$. This is not necessarily true for differential entropy.
Example. Let $a \neq 0$ be constant, and let X have differential entropy $h(X)$. Then $Y = aX$ has density

$$f_Y(y) = \frac{1}{|a|} f_X\left(\frac{y}{a}\right)$$

and

$$h(Y) = h(aX) = -\int \frac{1}{|a|} f_X\left(\frac{y}{a}\right) \left\{ \log f_X\left(\frac{y}{a}\right) - \log|a| \right\} \mathrm{d}y$$

$$= h(X) + \log|a|$$

And, more generally, if $\mathbf{X}_n$ is a random n-vector, and A an $n \times n$ matrix with non-zero determinant $detA$, then

$$h(A\mathbf{X}_n) = h(\mathbf{X}_n) + \log|detA|$$

1.10 Interpretation of entropy and information

While not essential in mathematics, interpretations of mathematical concepts are usually welcome, because they lend plausibility to axioms, and suggest which theorems should be most interesting. So we note that the concepts of entropy and mutual information defined above can be interpreted as measures of our real-world concepts of uncertainty, surprise, and information. We argue as follows:

(a) We defined the entropy $H(X)$ of the random variable X having probability distribution $p(x)$ to be the expected value of the empirical log-likelihood: $H(X) = E\{-\log p(X)\}$. Here is an intuitive interpretation of $H(X)$. Suppose that E is some event that may occur with probability $p = P(E)$. In advance of the relevant experiment we have some level of uncertainty about whether E will occur or not, and if later informed that E has occurred we feel some measure of surprise. The key point is that both our uncertainty and surprise vary according to $P(E)$. To see this consider E and E^c with probabilities $P(E) = 10^{-6}$, and $P(E^c) = 1 - 10^{-6}$. We feel rather more uncertain about the occurrence of E than E^c, and equally we would be rather more surprised at the occurrence of E than E^c. Since it is the transfer of information that has resolved the uncertainty and created surprise, all of these depend on $P(E) = p$. We claim that the following are intuitively natural properties of the surprise $s(E)$ that we feel about E's occurrence.

(i) It depends only on p, and not further on the value of any random variable defined on E, nor on any meaning conveyed by E, nor any other semantic aspect of E.
That is to say, $s(E) = u(p)$, for some function $u(p),\ 0 \le p \le 1$, taking numerical values.
For example, consider the events
E_1 = you win £10^6 with probability 10^{-6}.
E_2 = you are struck by lightning with probability 10^{-6}.
Obviously your feelings, (semantic connotations), about these two events, and the random outcomes defined on them, are very different. But you are equally surprised in each case.

(ii) The function $u(p)$ is decreasing in p. That is to say, you are more surprised by more unlikely events when they occur.

(iii) The surprise occasioned by the occurrence of independent events is the sum of their surprises. That is to say, if A and B are independent with probabilities p and q, then
$$s(A \cap B) = u(pq) = u(p) + u(q).$$

(iv) The surprise $u(p)$ varies continuously with p.

(v) There is no surprise in a certain event, so $s(\Omega) = u(1) = 0$.

From these it follows (by some analysis which we omit) that for some constant $c > 0$

$$u(p) = -c \log p$$

It is customary to take $c = 1$, and logarithms to base 2. Thus the unit of surprise is that which we experience on being told that a flipped fair coin showed a head. It is therefore also the amount of uncertainty about the event that a coin to be flipped will show a head. It follows that it is also the amount of information that we obtain with the news that a flipped fair coin showed a head, and this gives the canonical name for it: one bit of information.
Now for a simple random variable X, the surprise in any outcome $\{X = x\}$ is

$$-\log P(X = x) = -\log p(x)$$

Thus the expected surprise to be experienced when the actual value of X is revealed is

$$H(X) = -\sum_x p(x) \log p(x)$$

As above, this is also the average uncertainty about X before it is determined, and the expected amount of information to be obtained by discovering X. Likewise the pointwise conditional entropy $H(X \mid Y = y)$ is simply the surprise expected to be experienced on discovering X, given that we already know $Y = y$. And thus $H(X \mid Y)$ is the expected surprise on discovering X, after having been told Y, but before we know what either random variable is.
There are many other ways of justifying $H(X)$ as a measure of uncertainty and information; for example it is straightforward to write down (as Shannon did) a list of reasonable properties to be satisfied by $H(X)$ seen as a function of $p(x)$, $x \in A$. It then transpires that the only function consistent with these constraints is $H(X)$ as defined above. We omit this.

(b) We defined the mutual information $I(X;Y)$ as the Kullback-Leibler divergence between the joint distribution of X and Y, $p(x, y)$; (seen as the input and output of a channel), and the distribution $p_x(x)p_Y(y)$, where $p_X(x)$ and $p_Y(y)$ are the marginal distributions of $p(x, y)$. We then derived several representations for $I(X;Y)$ in terms of various entropy functions. Interestingly, each of these has an interpretation in terms of our intuitive ideas about the passage of information through a channel.

(i) As remarked above, $H(X)$ is the information that we seek to send, and $H(X \mid Y)$ is the expected remaining uncertainty about X felt by the receiver after the signal is transmitted. The difference

$$I(X;Y) = H(X) - H(X \mid Y)$$

is naturally interpreted as the amount of information successfully passed through the channel.

(ii) Likewise, $H(Y)$ is the information in the received signal, and $H(Y \mid X)$ is the noise that is induced in the original sent signal X. The difference is the information about X successfully transmitted, so

$$I(X;Y) = H(Y) - H(Y \mid X)$$

(iii) Finally, $H(X) + H(Y)$ is the total uncertainty in the sent and received signals separately, whereas $H(X, Y)$ can be seen as the uncertainty that they have in common. The difference is interpreted as the information passed

$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$

Alternatively, we can interpret $I(X;Y)$ through our intuitive ideas of surprise, as we did for the entropy. Let A_j be the event that a symbol of the signal is x_j, and B_k the event that the corresponding symbol of the received signal is y_k. Before the channel is used, the surprise to be occasioned in the receiver by the event A_j is

$$-\log P(A_j) = -\log p(x_j)$$

But conditional on B_k, that is the reception of the symbol y_k, the surprise occasioned by learning that actually x_j was sent will be

$$-\log P(A_j \mid B_k) = -\log p(x_j \mid y_k)$$

The difference in these levels of surprise is seen as the information passed by the channel in this event, i.e.,

$$I(x_j; y_k) = -\log p(x_j) + \log p(x_j \mid y_k) = \log \frac{p(x_j, y_k)}{p_X(x_j) p_Y(y_k)}$$

Before we use the channel, the expected value of this is interpreted as the information the channel expects to pass, i.e.,

$$\sum p(x_j, y_k) I(x_j; y_k) = I(X;Y)$$

as defined above.

2 Source coding

Shannon's first (source) coding theorem is an existence proof; that is to say, it tells us that there are compact codes that can compress messages arbitrarily close to the entropy lower bound. In this section, we discuss how to find optimal codes, and their implementation.

2.1 Compact symbol codes

Recall that, given a source emitting $X \in A$, a code is a map from the symbols of A to the set B^* of finite strings of symbols from B. Very often, but not always, $B = \{0, 1\}$. [For any n, we may encode the nth extension of the source by a map from A^n to B^*.] For $x \in A$, the codeword $c(x)$ has length $|c(x)|$, and the concatenated encoding of a message $\mathbf{x}$ is $c(\mathbf{x}) = c(x_1)c(x_2)\dots c(x_n)$. The expected length of such an encoding is

$$EL = E|c(\mathbf{X})| = n \sum_{x \in A} p(x)|c(x)|$$

Any code minimizing EL over choices of $c(\cdot)$ is said to be optimal or compact. Obviously desirable properties are these:

(a) A code is unambiguous if no two distinct source symbols are ever given the same codeword, i.e. $x \neq y \Rightarrow c(x) \neq c(y)$.

(b) A code is **uniquely decipherable** (or invertible) if any finite string from B^* is the image of at most one message from the source.

(c) A code is **prefix** (or prefix-free), or **instantaneous** if no codeword is the prefix of any other. That is to say for no x, y and string of symbols b^* from B^* is it the case that $c(x)b^* = c(y)$.

Example. Roll a die and encode $\{1, 2, 3, 4, 5, 6\}$ by their binary representations $\{1, 10, 11, 100, 101, 110\}$. This is unambiguous, but not uniquely decipherable as it is not prefix. Just consider how you might decode 110110110.
Example. An unambiguous block code is prefix, and therefore uniquely decipherable.
Example. Let $\{a, b, c\}$ be encoded by $\{0, 01, 11\}$. This is uniquely decipherable, though not prefix, but if the encoded message begins 0110, you must wait for further symbols to decipher it uniquely.
Note that although we have defined a code to be a map, we will often think of it equivalently as the collection of codewords, called the codebook. There is another, equally useful representation in terms of trees, defined thus:
Definition. A rooted **tree** is a connected acyclic graph, (that is, a collection of nodes (or vertices) joined by edges), with one node identified as the root, and all edges directed away from the root. It is said to be q-ary if at most q edges are directed away from any internal node; of course, external nodes have just one edge directed to them and none leaving. These are called **leaves**.
Then any q-ary prefix code may be identified with a q-ary rooted tree, in which the leaves of the tree correspond to the codewords. This may be called the codetree; and the representation is demonstrated by the codetree for the 3-symbol binary prefix code $\{0, 10, 11\}$; which has a root, one other internal node, and three leaves. You should sketch

this.
Note that in the binary case the number of leaves equals the number of internal nodes (including the root) plus one. And in a block code, all leaves are at the same distance from the root. The distance of a leaf from the root may be called its **height**, (and also, by some authors, its depth). A block q-ary tree having q^n leaves at height n is called the complete tree of height n. Leaves having the same parent node may be called **siblings**.
Now observe that the random source symbol X is encoded by the random codeword $c(X)$, which corresponds to a randomly selected leaf on the tree, which determines a unique random path $\pi(X)$ from the root to the leaf. [Because the tree is connected and acyclic.] This visualization is often useful, as for example in this theorem.
Theorem. For each internal node v in the codetree, let $\pi(v)$ be the collection of all paths $\pi(x_i),\ 1 \le i \le a$, that pass through v, and define

$$\sigma_v = \sum_{\pi(v)} p(x_i)$$

the sum of the probabilities of codewords on descendant leaves of the node v. Then

$$E|c(X)| = \sum \sigma_v$$

where the sum is over all internal nodes v, including the root.
Proof. Let I_v be the indicator of the event that the path $\pi(X)$ visits v. Then $EI_v = \sigma_v$, and

$$\begin{aligned} E|c(X)| &= E \sum I_v \\ &= \sum EI_v \\ &= \sum \sigma_v \end{aligned}$$

2.2 Prefix codes

The examples above, and the correspondence with trees, make it clear that prefix codes are very much preferable to the wider class of uniquely decipherable codes. Fortunately, it turns out that in seeking good compact codes we can confine our search to the class of prefix codes, as we now show. First, note that leaves on a tree identify a prefix code, whose word lengths $l_1, \ldots, l_n$ are equal to the height of the corresponding leaves on the tree. It is natural to ask, conversely, if some given collection of positive integers $l_1, \ldots, l_n$ can be the word lengths of a prefix code. The answer is given by this:

Theorem. Kraft's inequality.
If the positive integers $l_1, \ldots, l_n$ satisfy

$$(*) \qquad \sum_{r=1}^{n} 2^{-l_r} \le 1$$

then there exists a binary prefix code having $l_1, \ldots, l_n$ as its word-lengths.
Proof. Let $l_1 \le l_2 \le \cdots \le l_n$ be positive integers satisfying $(*)$, which we re-write as

$$(+) \qquad 2^{l_n - l_1} + 2^{l_n - l_2} + \cdots + 1 \le 2^{l_n}$$

Consider the complete tree of height l_n, with 2^{l_n} leaves at height l_n. Now place a leaf at any internal node C_1 height l_1; because we require a prefix code this excludes $2^{l_n-l_1}$ leaves, that are descendants of c_1 at height l_n, from consideration as part of the code. By $(+)$, $2^{l_n} - 2^{l_n-l_1} > 1$ leaves remain at height l_n.
We can thus place a leaf at an internal node c_2 at height l_2. Then $2^{l_n} - 2^{l_n-l_1} - 2^{l_n-l_2} > 1$ leaves still remain at height l_n. Continuing to the end of the sequence, by $(+)$ there will be a leaf at height l_n to yield the codeword of length l_n. The fact that we can confine our attention to codes with this property, justifying our claim above, follows from this next theorem:
McMillan's Theorem. Let $l_1, \ldots, l_n$ be the codeword lengths of a uniquely decipherable binary code. Then

$$\sum_{r=1}^{n} 2^{-l_r} \leq 1$$

Proof. Let N be an arbitrary integer, and let A_k be the number of ways in which N codewords can be concatenated to form a string of length k. Then, writing $l = max\{l_1, l_2, \ldots, l_n\}$, it is identically true that

$$(\neq) \qquad \left(\sum_{r=1}^{n} 2^{-l_r} \right)^N = \sum_{k=1}^{Nl} A_k 2^{-k}$$

Since the codewords form a uniquely decipherable code, we must have $A_k \leq 2^k$, because unique decipherability requires that the number of concatenations of codewords of length k is no greater than the number of k strings. Hence the right side of $(\neq)$ is no greater than Nl, and so

$$\sum_{r=1}^{n} 2^{-l_r} \leq (Nl)^{\frac{1}{N}} \to 1 \quad \text{as } N \to \infty$$

Hence, by Kraft's inequality, there is a prefix code with these word-lengths.
Corollary. A binary code with word lengths $l_1, \ldots, l_n$ that is prefix (i.e. instantaneous) exists if and only if

$$\sum_{r=1}^{n} 2^{-l_r} \leq 1$$

And to any uniquely decipherable code with word lengths $l_1, \ldots, l_n$, there corresponds a prefix code having the same word lengths.
Likewise, a q-ary code with word lengths $l_1, \ldots, l_n$ exists, that is prefix, if and only if

$$\sum_{r=1}^{n} q^{-l_r} \leq 1$$

2.3 The entropy bound for noiseless coding

After the preliminaries above, we can now prove another theorem of Shannon's:
Theorem: Noiseless coding
Let a discrete memoryless source have distribution $P(X = x_r) = p_r$, and entropy H.

(a) Then any uniquely decipherable binary code $c(X)$ for this source must satisfy

$$EL = E|c(X)| \geq H$$

with equality iff $l_r = |c(x_r)| = -\log p_r$.

(b) Furthermore, there is such a code such that

$$E|c(X)| \leq H + 1$$

Proof of (a). Define the probability distribution q_r on the alphabet A of X by

$$q_r = 2^{-l_r} / \{\sum_{r=1}^{a} 2^{-l_r}\}$$

Then

$$\begin{aligned} E|c(X)| - H(X) &= \sum p_r l_r + \sum p_r \log p_r \\ &= -\sum p_r \log 2^{-l_r} + \sum p_r \log p_r \\ &= \sum_r p_r \log \frac{p_r}{q_r} - \sum_r p_r \log \left(\sum_k 2^{-l_k} \right) \\ &= d(p,q) - \log \left(\sum 2^{-l_k} \right) \geq 0 \end{aligned}$$

by Gibbs inequality, and the fact that $\sum 2^{-l_k} \leq 1$, because the code is uniquely decipherable. Equality holds iff we have both $\sum 2^{-l_k} = 1$ and $p_r = q_r = 2^{-l_r}$ for all r.
In this case we must have that $-\log p_r$ is an integer, and a distribution of this form is said to be 2-adic, (or, by some authors, dyadic).
Example. The distribution $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$ is 2-adic, and the optimal encoding $\{0, 10, 110, 111\}$ has expected word length $1.75 = H(X)$.

Proof of (b). Now consider the possibility of a binary code with word lengths

$$s_r = \lceil -\log p_r \rceil, \qquad 1 \leq r \leq a$$

where $\lceil x \rceil$ is the smallest integer not less than x; (often called the ceiling). Then

$$\sum 2^{-s_r} \leq \sum 2^{-(-\log p_r)} = \sum p_r = 1$$

Hence, by the Kraft-McMillan theorems, there is a prefix code (called the Shannon code) with word lengths s_r. And by construction

$$-\log p_r \leq s_r < -\log p_r + 1;$$

so that multiplying by p_r and summing over r, gives

$$H(X) \leq EL < H(X) + 1$$

for EL the expected codeword length, per source symbol, of this code.
Remark. The difference $R = EL - H(X)$ is called the **redundancy** of ths code.

Corollary 1. The encoding redundancy (per source symbol) of a DMS can be made as small as we please as the message length increases.

Proof. Concatenate the source letters in blocks of length n, and encode the blocks with a Shannon prefix code (or anything better). Then by the above, in an obvious notation,

$$\begin{aligned} H(X_1, \ldots, X_n) = nH(X) \leq EL_n < nH(X) + 1 \\ = H(X_1, \ldots, X_n) + 1 \end{aligned}$$

Dividing by n, the redundancy of the code per source symbol, namely $\frac{1}{n}EL_n - H$, is less than n^{-1}. Hence

$$\frac{1}{n}EL_n - H \to 0, \text{ as } n \to \infty$$

Corollary 2. Let $X \in A$ be a simple random variable whose value is known to the oracle. You know A, and your task is to identify the value of X by asking a sequence of questions, to which the oracle's answer can only be either yes or no. Then the minimum expected number Q of questions required satisfies $H(X) \leq Q < H(X) + 1$.
Proof Setting Yes $\equiv 1$ and No$\equiv 0$, any successful strategy of interrogation corresponds to a binary prefix code for X; (and thus also to a tree). The result follows from what we have shown above.
Example. In Cambridge it is known that an outbreak of gastric infection originates in the kth of the n possible restaurants with probability p_k. Samples from each can be pooled and tested, so that for any pool R it will be known if the source is in R or not. The minimum expected number I of such tests required to identify the source of the outbreak obeys

$$H(\mathbf{P}) \leq \tau < H(\mathbf{P}) + 1$$

Example. Of n superficially identical coins, just one (or none) may be a forgery that is either heavy or light. You have a balance that will accept any two sets of coins, and tell you whether one set is heavier than, lighter than, or of equal weight to, the other set. The minimum expected number of weighings is $\log_3(2n+1)$, because there are $2n+1$ possible cases, assumed to be equally likely. And, remarkably, for $n = 12$, three weighings are sufficient.

The natural question remains, is there an optimal (compact) code? The answer is yes, as we see next.

2.4 Optimality: Huffman codes

In this section we give Huffman's procedure for constructing a compact prefix code for a DMS, and then prove that it is indeed optimal. For simplicity, we confine ourselves to binary codes. [Note that there may be other codes that perform equally well, that cannot

be produced by Huffman's procedure.]

Huffman's binary code
The essential idea of the procedure is to construct a binary tree that yields a prefix code with shortest expected codeword length. The clever feature of the procedure is that the tree is constructed not from the root, but from the topmost pair of leaves. Thus, let $p_1 \geq p_2 \geq \cdots \geq p_a, \qquad a = |A|$.
Combine the two smallest probabilities; this corresponds to forming a node of the tree at the highest level; with leaves corresponding to the codewords for the symbols with probabilities p_{a-1} and p_a.
Now reorder the resulting distribution to give

$$p'_1 \geq p'_2 \geq \cdots \geq p'_{a-1},$$

where $p'_{a-1} = p_a + p_{a-1}$.
Combine the two smallest probabilities, corresponding to the formation of the next highest node of the tree. Continuing in this way we ultimately arrive at unit probability, which corresponds to the root of the tree. Then each external vertex of the tree carries a leaf, which may be labelled with the appropriate binary codeword, traced up from the root.
This procedure is most easily understood through an example and an obvious diagram.
Example. Consider the source distribution

$$\mathbf{p} = (0.5, 0.25, 0.15, 0.05, 0.05)$$

Combining the two smallest yields

$$\mathbf{p}' = (0.5, 0.25, 0.15, 0.1)$$

and then

$$\mathbf{p}'' = (0.5, 0.25, 0.25)$$
$$\mathbf{p}''' = (0.5, 0.5)$$

, and finally

$$\mathbf{p}^{(iv)} = 1$$

, at the root.
Representing this as a diagram yields a tree, with leaves at the vertices where probabilities are first combined; you should sketch this here.

Labelling the leaves according to the natural correspondence, we obtain the codewords for the symbols with probabilities $p_r,\ 1 \leq r \leq a$

$$p_1 \to 1$$
$$p_2 \to 01$$
$$p_3 \to 001$$
$$p_4 \to 0001$$
$$p_5 \to 0000$$

Thus

$$E|c(X)| = 0.5 + 0.5 + 0.45 + 0.2 + 0.2$$
$$= 1.85$$

codeword symbols per source symbol. A slightly tedious calculation gives $H(X) \doteqdot 1.84$, so the Huffman code is very close to the bound. This labelling (in reverse, from the right) is the Huffman expansion.
We now prove that this procedure always supplies an optimal code for a source X with symbol probabilities $p_1 \geq p_2 \geq \cdots \geq p_a > 0$. Before the principal result, we need these results, in which (as usual) $(l_r,\ 1 \leq r \leq a$ are word lengths:)

Lemma.

(a) For such a source, as defined above, there is an optimal code.

(b) For any optimal code

(i) if $p_j \geq p_k$, then $|c(j)| = l_j \leq l_k = |c(k)|$

(ii) The two (or more) longest codewords have the same length

(iii) There is a code with the same value of $E|c(X)|$ such that two of the longest codewords correspond to the two least likely symbols, and are siblings.

Proof

(a) There is at least one binary prefix code for X, namely a block code having more leaves than a. And only a finite number of prefix codes for X have expected codeword length less than this block code, so one of them, or the block code, is optimal. To see this, enumerate the codes in order of increasing maximum codeword length l_a. Eventually $p_a l_a$ exceeds the block code length.

(b) (i) For such an optimal code exchange the codewords for the symbols j and k, having respective probabilities and codeword lengths (p_j, l_j) and (p_k, l_k). The difference in expected codeword lengths for the two codes is

$$p_j l_j + p_k l_k - p_j l_k - p_k l_j = (p_j - p_k)(l_j - l_k) \quad \leq 0$$

since the original code was assumed optimal. Since $p_j \geq p_k$, it follows that $l_j \leq l_k$

(ii) If one codeword were longer than the rest, the last (highest) symbol (edge) could be deleted giving a prefix code with shorter expected word length, which contradicts the assumption of optimality

(iii) If one of the longest codewords has no sibling, then it can be shortened as in (ii). Thus all the longest codewords have siblings, and these can be arranged (without altering the expected length) so that the two lowest probability codewords are siblings.

The code with the above properties is called the canonical optimal (or compact) code.
And the point of this construction is that at each stage of the Huffman algorithm it is the two least likely sibling codeword leaves that are combined to give a code or an alphabet with one fewer symbols, and conversely.

Theorem. The binary Huffman procedure yields an optimal code.

Proof. Let $p_m \leq p_{m-1} \leq \cdots \leq p_1$ and $l_m \geq l_{m-1} \geq \cdots \geq l_1$ be the probabiltiies and codeword lengths at any stage in the Huffman procedure. That is to say, in the reduction procedure $p_m + p_{m-1}$ is the probability of the new leaf for a code on $m-1$ symbols, with expected codeword length c_{m-1}, and in the expansion procedure the leaf with probability $p_m + p_{m-1}$ is replaced by a parent node with sibling leaves of probability p_m and p_{m-1}, for a code on m symbols with expected codeword length c_m

Assume that the code on m symbols is optimal and canonical, (as we may, by the Lemma above), with expected length b_m. Then by the final Lemma of 2.1, after reduction

$$b_m = c_{m-1} + p_m + p_{m-1}$$

Now conversely, suppose that the optimal code on $m-1$ symbols (with probabilities $p_m + p_{m-1}, p_{m-2}, \ldots, p_1$) has expected length b_{m-1}. After expansion (using the Lemma in 2.1 again) we have

$$c_m = b_{m-1} + p_m + p_{m-1}$$

Hence, subtracting,

$$0 \leq c_m - b_m = b_{m-1} - c_{m-1} \leq 0$$

where the inequalities follow from the assumed optimality of b_m and b_{m-1} for each alphabet.

Hence $b_m = c_m$, and $b_{m-1} = c_{m-1}$; which is to say that at any stage of the Huffman procedure, optimality of the encoding is preserved when the alphabet is either reduced or expanded. But the encoding $[0, 1]$ of the alphabet of the two symbols is clearly optimal. Hence the Huffman encoding of the alphabet of a symbols is also optimal.

A similar procedure is used to produce Huffman codes in a q-ary encoding alphabet, with the proviso that the alphabet of source symbols may need to be augmented with dummy symbols of zero probability, in order that the final Huffman reduction is to exactly q letters, (for the obvious optimal encoding).

2.5 Other prefix codes

The Huffman code is optimal, but it is often useful to consider other prefix codes which, while not optimal, are more quickly generated, or more tractable to analysis in various contexts. One of the simplest is this:

Example. Fano code.

A simpler but sub-optimal method of forming a prefix code is this: order the symbols by their probability. Divide them into two groups with respective probabilities as nearly equal as possible; i.e., more formally, find r such that

$$\left| \sum_{i=1}^{r} p_i - \sum_{r+1}^{a} p_i \right|$$

is minimal. For those in the first group the first digit in the codeword is 0; for those in the second group it is 1. Continuing in this way until each group contains just one symbol generates a prefix code. It can be shown, (as a corollary of the final theorem in this section), that $E|c(X)| \leq H(X) + 1$; (or, if you do not trouble to order the symbols by their probability, $E|c(X)| \leq H(X) + 2$).

Example. Shannon code.
Again, order the probabilities, and then define

$$F_r = \sum_{i=1}^{r-1} p_i,\ 1 \leq r \leq a,$$

where the empty sum is zero. The binary code for x_r is then the binary expansion of F_r, carried as far as the l_rth place, where l_r is the smallest integer not less than $\log p_r^{-1}$.To see that this produces a prefix code, note that for $r \geq k$,

$$F_r - F_{k-1} \geq F_k - F_{k-1} = p_{k-1} \geq 2^{-l_{k-1}}$$

[because $l_{r-1} \geq -\log p_{r-1}$], and therefore F_r must differ from F_{k-1} in at least one of the first l_{k-1} binary digits of F_{k-1}. So no codeword is the prefix of another. Since

$$-\log p_r \leq l_r < -\log p_r + 1$$

we have

$$E|c(X)| < H(X) + 1$$

Example. Elias code. This can be seen as a variant of the Shannon code. Assume that $p_r > 0,\ r \in A$, and define

$$R_k = \sum_{i<k} p_i + \frac{1}{2} p_k,\ 1 \leq k \leq a$$

Draw a diagram to see that the points R_k are midway up the jumps of the cumulative distribution function of $\{p_r,\ r \in A\}$.
Then the Elias code for x_k is the binary expansion of R_k, truncated at the l_kth place, where l_k is the ceiling of $-\log p_k$ plus 1; i.e.,

$$-\log p_k + 1 \leq l_k = \lceil -\log p_k \rceil + 1 < -\log p_k + 2$$

Thus $2^{-l_k} < \frac{1}{2} p_k$, and the codeword $c(x_k)$ lies in the same jump as R_k. Furthermore, the code is prefix free by an argument similar to that of the preceding example.

Example. Fix-free code
A prefix code in which codewords reversed form an instantaneous, or prefix-free, code.

Since any prefix code is equivalent to a suitable tree, with a probability distribution $(p_1, \ldots, p_a)$ on its leaves, we now develop a form of the entropy bound that exploits this structure. We consider binary codes and trees, for simplicity.

First, recall the theorem in §2.1 in which we showed that $E|c(X)| = \sum \sigma_v$, where σ_v is the probability that the path $\pi(X)$ from the root to the leaf corresponding to the codeword $c(X)$ passes through the internal node v.
Second, note that we can generate the same distribution of codewords and their probabilities on the leaves of the tree by realizing a random walk from the root that steps either left or right at any internal node, with the following distribution:
let ρ_v and λ_v be the probabilities attached to the two daughter nodes of v, so that

$$\sigma_v = \rho_v + \lambda_v$$

and in particular for the root $\sigma_1 = 1 = \rho_1 + \lambda_1$.

Then the step from v is conditionally independent of its route from the root (i.e., given that the path is at v), and steps right with probability ρ_v/σ_v or left with probability λ_v/σ_v. Of course, the walk stops on reaching any leaf, and for a given leaf, labelled $c(x)$, the probability that the walk arrives there via the sequence of vertices $v(1), v(2), \ldots, v((x)-1)$ is

$$\sigma_v(1)\frac{\sigma_v(2)}{\sigma_v(1)}\frac{\sigma_v(3)}{\sigma_v(2)}\cdots\frac{\sigma_v(l(x)-1)}{\sigma_v(l(x)-2)}\frac{p(x)}{\sigma_v(l(x)-1)} = p(x)$$

as claimed.
Naturally, for any vertex v that $\pi(x)$ does not visit, no action is required. Thus, if we number the internal nodes sequentially from the root, by height (and indifferently between those at the same height), then we can attach an auxiliary random variable A_v to each node, with the following properties:

$$A_v \in \{\text{left,right,null}\}$$

where $P(A_v = \text{null}) = 1 - \sigma_v$, and $P(A_v \in \text{ left } \cup \text{ right }) = \sigma_v$. More importantly, we may consider the distribution of A_v conditional on $\{A_1, \ldots, A_{v-1}\}$. If these entail that v is visited, then the conditional distribution of A_v is $\{\lambda_v/\sigma_v, \rho_v/\sigma_v, 0\}$ with entropy $H(\lambda_v/\sigma_v, \rho_v/\sigma_v)$. Otherwise, the conditional distribution of A_v is $[0, 0, 1]$, with entropy 0. Hence the conditional entropy is

$$(*) \quad H(A_v \mid A_1, \ldots, A_{v-1}) = \sigma_v H(\lambda_v/\sigma_v, \rho_v/\sigma_v) = \sigma_v H_v$$

Finally, we note that

$$(+) \quad H(X) = H(c(X)) \qquad \text{, by invertibility}$$

$$= H(A_1, \ldots, A_{a-1})$$

because, by construction, the value of $c(X)$ determines $\pi(X)$, and hence all the A_vs, and conversely.
With these facts, we can prove this
Theorem. Entropy bound for prefix codes
The redundancy of a prefix code is

$$0 \le R = E|c(X)| - H(X) = \sum_v \sigma_v(1 - H_v)$$

$$\leq \sum_v |\lambda_v - \rho_v|$$

with equality throughout if $(p_1, p_2, \ldots, p_a)$ is 2-adic, and where the sum is over the internal nodes of the tree.

Proof By the chain rule, and $(*)$ and $(+)$ above,

$$H(X) = H(A_1) + H(A_2 \mid A_1) + \cdots + H(A_{a-1} \mid A_1, \ldots, A_{a-2})$$

$$= \sigma_1 H_1 + \sigma_2 H_2 + \cdots + \sigma_{a-1} H_{a-1}$$

so that

$$R = \sum_v \sigma_v - H(X) = \sum_v (\sigma_v - \sigma_v H_v)$$

For the final inequality, observe (by drawing a picture) that

$$1 - H(\rho, 1-\rho) \leq |1 - 2\rho|, \ 0 \leq \rho \leq 1$$

so

$$R \leq \sum_v \sigma_v |1 - 2\rho_v/\sigma_v| = \sum_v \sigma_v |\lambda_v/\sigma_v - \rho_v/\sigma_v|$$

$$= \sum_v |\lambda_v - \rho_v|$$

with equality if $\lambda_v = \rho_v$ at every internal node.

The term $\sigma_v(1 - H_v)$ may be called the local redundancy.

3 Channel capacity and noisy coding

In our sketch of the direct part of Shannon's noisy coding theorem, in §1.8, we showed that arbitrarily reliable transmission of messages was possible up to the rate C, where C is the capacity of the channel. In this chapter we find the capacity of several important channels, and establish some properties of the capacity. Finally, we give an alternative proof of the noisy coding theorem, together with the converse.

3.1 Introduction: basic channels

Recall that a discrete memoryless channel is characterized by its channel matrix $M = p(y \mid x)$, where $X \in A$ and $Y \in B$ denote the input and output respectively. We introduce some vocabulary:

Lossless. A channel is a lossless if $H(X \mid Y) = 0$ always. That is to say the alphabet of Y can be divided into disjoint sets U_i such that

$$P(Y \in U_i \mid X = x_i) = 1,\ 1 \leq i \leq a = |A|$$

Deterministic A channel is deterministic if $H(Y \mid X) = 0$, always. That is to say $p(y \mid x)$ is either 0 or 1, for all x and y.

Perfect A channel is perfect (or noiseless) if it is lossless and deterministic. Its channel matrix is then the identity matrix; or, equivalently, a permutation of the identity matrix.

Useless A channel is useless if Y is independent of X.

Recall also that the capacity is $C = \max_{p_X(x)} I(X;Y)$.

Hence we have this

Lemma

$$C \leq \log(\min\{|A|,\ |B|\})$$

Proof Follows immediately from the identities

$$I(X;Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$

and the fact that $H(X) \leq \log a$ for any entropy $H(X)$ with alphabet size a.

Erasure Channel. One in which $|B| = |A| + 1 = a + 1$, and one column of the channel matrix is constant; that is $P(y_{a+1} \mid X) = \epsilon > 0$ for all x, may be called an a-ary erasure channel, because of the natural way in which the matrix arises from the possible erasure of input symbols by the channel.

Example. Binary channels

In general this has channel matrix of the form

$$M = \begin{pmatrix} p & 1-p \\ q & 1-q \end{pmatrix}, \quad 0 \leq p,\ q \leq 1$$

If $p + q = 1$, this is called the **binary symmetric channel** or BSC; and if $p = 1$, and $0 < q < 1$, it may be called the Z-channel.

If there is an extra constant column, so that

$$M = \begin{pmatrix} p & 1-p-r & r \\ 1-q-r & q & r \end{pmatrix}$$

then this is the **binary erasure channel**, (BEC). Almost always, it is assumed that $p = q$, giving the **binary symmetric erasure channel**, (BSEC).
The capacity of such binary channels is fairly easily found by elementary methods of calculus.
Example. BSC
Here

$$C = 1 - H(p, 1-p) = 1 + p \log p + (1-p) \log(1-p)$$

which we can demonstrate as follows. Let the distribution of X be $(x, 1-x)$. Then

$$I(X;Y) = H(Y) - H(Y \mid X)$$

$$= H(x + p - 2xp, 1 - x - p + 2xp) - H(p, 1-p)$$

Differentiating for x yields a stationary value, which is a maximum, when

$$\log(x + p - 2xp) = \log(1 - x - p + 2xp)$$

so that $x = \frac{1}{2}$. Hence

$$C = 1 - H(p, 1-p)$$

Alternatively, if the distribution of X is written as (x, y), where $x + y - 1 = 0$, the method of Lagrange multipliers yields a maximum when

$$H(xp + y(1-p), yp + x(1-p)) + \lambda(x + y - 1) - H(p, 1-p)$$

has a stationary value, so that $x = y = \frac{1}{2}$, as above.
However, in cases such as these, where the channels have an appropriate symmetry, we can exploit that fact to yield the capacity of M by more elegant arguments.
We look at symmetric channels in the next section, in more detail.
We conclude this section with a useful result about the capacity of the nth extension of a channel.
Lemma. The capacity of the nth extension of a channel M with capacity C does not exceed nC.
Proof

$$I(\mathbf{X}; \mathbf{Y}) = H(\mathbf{Y}) - H(\mathbf{Y} \mid \mathbf{X})$$

$$= H(\mathbf{Y}) + E \sum_{r=1}^{n} \log p(Y_r \mid X_r)$$

by memorylessness

$$\leq \sum_{r=1}^{n} H(Y_r) - \sum_{r=1}^{n} H(Y_r \mid X_r) \quad \text{with equality iff } Y_r \text{ are independent}$$

$$= \sum_{r=1}^{n} I(X_r; Y_r)$$

and the result follows.

3.2 Symmetric channels

Certain symmetries in the channel matrix M enable us to find the channel capacity rather easily.

Definition M is said to be strongly symmetric if its rows are permutations of each other, and its columns are permutations of each other.

E.g.,

$$\begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}$$

More generally M is said simply to be symmetric if its rows are permutations of each other, and the column sums of M are the same. E.g.,

$$\begin{pmatrix} \frac{1}{6} & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{3} \end{pmatrix}$$

For both these types we have this

Theorem. If $\boldsymbol{\rho}$ is any row of a symmetric channel matrix M, then its capacity is

$$C = \log b - H(\boldsymbol{\rho})$$

where $b = |B|$, the size of Y's alphabet; and this is achieved with a uniform distribution on X.

Proof

$$I(X;Y) = H(Y) - H(Y \mid X)$$

$$= H(Y) - H(\boldsymbol{\rho}) \text{ by the symmetry}$$

$$\leq \log b - H(\boldsymbol{\rho})$$

since $H(Y) \leq \log |B|$, with equality if and only if Y is uniformly distributed. But if X is uniform on its alphabet, so $P(X = x) = a^{-1}$, then

$$p(y) = \frac{1}{a} \sum_{x=1}^{a} p(y \mid x) = \frac{c}{a} = \frac{1}{b}$$

where c is the column sum of M. Therefore Y is uniform on its alphabet also. Hence the upper bound $\log b - H(\boldsymbol{\rho})$ is attained for uniform X, and this is the capacity.

More generally, we may consider this

Definition A channel M is weakly (or generally) symmetric if the output (that is to say, the columns of M) can be grouped into subsets, yielding submatrices M_i of M, such that for each i the matrix M_i is symmetric in the sense of the previous definition. That is:-

(a) The rows of M_i are permutations; (and hence so also are the rows of M)

(b) Each row of M_i has κ_i entries with sum ρ_i

(c) Each M_i has column sum that is constant c_i

Then we have the following result for such a channel:-
Theorem For a weakly symmetric channel with row entropy $H(\mathbf{p}) = -\sum p_i \log p_i$; the capacity is

$$C = \sum \rho_i \log \kappa_i - \sum \rho_i \log \rho_i - H(\boldsymbol{p})$$

which is achieved by a uniform distribution on the input X.
Proof. Denoting the output by Y, define the random variable $J = J(Y)$, which takes the value i if Y corresponds to a column of M_i. Note that $H(J) = H(J \mid X)$, because the rows of each M_i have the same sum ρ_i, so J and X are independent. Also, as J is a function of Y

$$H(Y) = H(J, Y) = H(J) + H(Y \mid J),$$

and

$$H(Y \mid X) = H(J, Y \mid X) = H(J \mid X) - H(Y \mid J, X),$$

and

$$\begin{aligned} C &= \sup_{p_X(x)} (H(Y) - H(Y \mid X)) \\ &= \sup_{p_X(x)} [H(Y \mid J) - H(Y \mid J, X)] \end{aligned}$$

But for each i, on the event $J = i$, the term $H(Y \mid J = i) - H(Y \mid J = i, X)$ achieves its maximum (by the argument above) when X is uniform. Therefore, since $P(J = i) = \rho_i$, and denoting the entries of a row of M_i by $p_r(i)$, $1 \le r \le \kappa_i$, we have

$$C = \sum_i \rho_i (\log \kappa_i + \sum_{r=1}^{\kappa_i} (p_r(i)/\rho_i) \log(p_r(i)/\rho_i))$$

$$= \sum_i \rho_i \log \kappa_i - H(\boldsymbol{p}) - \sum_i \sum_{r=1}^{\kappa_i} p_r(i) \log \rho_i$$

as asserted, because

$$\sum_{r=1}^{\kappa_i} p_r(i) = \rho_i$$

Example. Consider the binary erasure channel

$$\begin{pmatrix} 1-a & 0 & a \\ 0 & 1-a & a \end{pmatrix}$$

where $\kappa_1 = 2,\ \kappa_2 = 1,\ \rho_1 = 1 - a,\ \rho_2 = a$. Then the above theorem supplies the capacity $C = 1 - a$, as shown above and below by other methods.

3.3 Special channels

Here we consider a number of important special channels whose capacity can be found explicitly.

Erasure channel. Arises when any symbol of B is independently deleted by the channel, with constant probability β, and this is known to the receiver. It is thus customary to treat this as a channel with output alphabet $B \cup *$, where $*$ represents erasure. Any such channel can be seen as equivalent to this formulation:-

A channel M has capacity C, input $X \in A$ and output $Y \in B$. The output symbols are each independently erased with probability β, or passed with probability $1-\beta$, yielding an output $Z \in B \cup *$. We shall show that the capacity of the composite channel, with input X and output Z, is $(1-\beta)C$, where this is achieved by the input distribution of X that achieves the capacity C for M.

Proof. Let J be the indicator of the event that a symbol is erased, so that $P(J=1) = EJ = \beta$. We note that J is independent of X, and also a function of Z. Now by the chain rule

$$\begin{aligned} H(Z) &= H(Z,J), \text{ because } J \text{ is a function of } Z \\ &= H(J) + (1-\beta)H(Z \mid J=0) + \beta H(Z \mid J=1) \\ &= H(J) + (1-\beta)H(Y), \text{ since } Z = Y \text{ on } \{J=0\}, \text{ and } H(Z \mid J=1) = 0 \end{aligned}$$

Likewise

$H(Z \mid X) = H(Z, J \mid X)$ since J is independent of X, and a function of Z,

$$\begin{aligned} &= H(J \mid X) + (1-\beta)H(Z \mid X, J=0) + \beta H(Z \mid X, J=1) \\ &= H(J) + (1-\beta)H(Y \mid X), \text{ since } Z = Y \text{ on } \{J=0\}, \text{ and } H(J|X) = H(J) \end{aligned}$$

Hence

$$\begin{aligned} I(X;Z) &= H(Z) - H(Z \mid X) \\ &= (1-\beta)(H(Y) - H(Y \mid X)) \\ &= (1-\beta)I(X;Y) \end{aligned}$$

and the result follows, on taking the supremum over all input distributions

Channels in series: the data-processing theorem

Let X and Y be the input and output respectively of a channel M with matrix $p_1(y \mid x)$. Let N be a second channel, independent of M, whose input alphabet is the same as that of Y. The output Y is now entered as the input of N, with output Z. Show that $I(X;Y) \geq I(X;Z) \leq I(Y;Z)$.

Proof. Let N have matrix $p_2(z \mid y)$. Then

$$\begin{aligned} p(x, z \mid y) &= p(x,y,z)/p(y) = p(x,y)p(z \mid x, y)/p(y) \\ &= p(x \mid y)p_2(z \mid y) \end{aligned}$$

because Z is independent of X given Y, by the independence of the channels. Hence $I(X; Z \mid Y) = 0$, and we can use the chain rule to write

$$I(X;Y) = I(X;Y,Z) - I(X;Z \mid Y)$$

$$= I(X; Y, Z) = I(X; Z) + I(X; Y \mid Z)$$

and the first result follows since $I \geq 0$.
The second follows in exactly the same way. This result can be interpreted as showing that no form of processing or statistical manipulation of the output of a channel can increase its capacity. These inequalities may equivalently be written as

$$H(X \mid Y) \leq H(X \mid Z) \geq H(Y \mid Z)$$

Sum of channels
The sum of two channels has channel matrix

$$M = \begin{pmatrix} M_1 & 0 \\ 0 & M_2 \end{pmatrix}$$

where M_1 and M_2 are channel matrices. It may be seen as the compound channel arising when one may choose to use either one of two independent channels, with disjoint input alphabets and disjoint output alphabets. The capacity C of such a channel is given by

$$2^C = 2^{C_1} + 2^{C_2}$$

where C_1 and C_2 are the capacities of M_1 and M_2 respectively.
Proof Let the distribution of the input X be $\theta\mathbf{p}$ over the alphabet of M_1, and $(1-\theta)\mathbf{q}$ over the alphabet of M_2, for probability distributions $\mathbf{p}$ and $\mathbf{q}$. Let J be the indicator of the event that M_1 is used, so that

$$P(J = 1) = \theta = 1 - P(J = 0)$$

Note that J is a function of X, and also a function of Y. Therefore $H(J \mid X) = 0$ and

$$I(X; Y) = H(Y) - H(Y \mid X)$$

$$= H(Y, J) - H(Y, J \mid X)$$

$$= H(J) - H(J \mid X) + \theta\{H(Y \mid J = 1) - H(Y \mid X, J = 1)\} + (1-\theta)\{H(Y \mid J = 0) - H(Y \mid X, J = 0)\}$$

Choosing $\mathbf{p}$ and $\mathbf{q}$ to maximise the coefficients of θ and $1 - \theta$ respectively, we then have

$$C = \max_{\theta}[H(J) + \theta C_1 + (1 - \theta)C_2]$$

Since

$$H(J) = -\theta \log \theta - (1 - \theta) \log(1 - \theta), \text{ so that}$$

$$\frac{\partial I}{\partial \theta} = \log\left(\frac{1-\theta}{\theta}\right) + C_1 - C_2,$$

$$\text{and } \frac{\partial^2 I}{\partial \theta^2} = \frac{-1}{\theta(1-\theta)} < 0,$$

yielding a maximum. Hence we find that the required value of θ is

$$\theta = \frac{2^{C_1}}{2^{C_1} + 2^{C_2}}, \text{ and } 1 - \theta = \frac{2^{C_2}}{2^{C_1} + 2^{C_2}}$$

and substitution yields $C = \log(2^{C_1} + 2^{C_2})$ as asserted.

Product of channels. Let $\mathbf{X} = (X_1, X_2)$ and $\mathbf{Y} = (Y_1, Y_2)$ be the input and output of two channels M_1 and M_2 with respective capacities C_1 and C_2, that are conditionally independent given their joint input $\mathbf{X} = (X_1, X_2)$. (Which is to say that $p(\mathbf{Y}|\mathbf{X}) = p_1(y_1|x_1)p_2(y_2|x_2)$.)Thus their joint distribution takes the form

$$P(\mathbf{X} = \mathbf{x}, \mathbf{Y} = \mathbf{y}) = P(\mathbf{X} = \mathbf{x})p_1(y_1 \mid x_1)p_2(y_2 \mid x_2)$$

Since Y_1 and Y_2 are conditionally independent given $\mathbf{X}$,

$$H(\mathbf{Y} \mid \mathbf{X}) = H(Y_1 \mid \mathbf{X}) + H(Y_2 \mid \mathbf{X})$$

Furthermore, Y_1 is conditionally independent of X_2 given X_1; and Y_2 is conditionally independent of X_1 given X_2. Hence $H(Y_1 \mid \mathbf{X}) = H(Y_1 \mid X_1)$ and $H(Y_2 \mid \mathbf{X}) = H(Y_2 \mid X_2)$. Thus

$$\begin{aligned} I(\mathbf{X}; \mathbf{Y}) &= H(\mathbf{Y}) - H(\mathbf{Y} \mid \mathbf{X}) \\ &= H(Y_1, Y_2) - H(Y_1 \mid X_1) - H(Y_2 \mid X_2) \\ &\leq H(Y_1) + H(Y_2) - H(Y_1 \mid X_1) - H(Y_2 \mid X_2) = I(X_1; Y_1) + I(X_2; Y_2), \end{aligned}$$

with equality if and only if Y_1 and Y_2 are independent. This occurs if X_1 and X_2 are independent. Hence

$$C \leq C_1 + C_2$$

where equality holds if X_1 and X_2 are independent, having the distributions that achieve capacity in M_1 and M_2 respectively. The composite channel is said to be the product of M_1 and M_2.

3.4 Concavity of H and I

A number of functions arising in information theory have properties of concavity or convexity that are often useful. We mention some important examples.

Theorem The entropy $H(\mathbf{p})$ is a strictly concave function of $\mathbf{p}$; that is to say, for $0 < t < 1$, and $\mathbf{p} \neq \mathbf{q}$, where $\mathbf{p}$ and $\mathbf{q}$ are distributions on the same alphabet, we have

$$H(t\mathbf{p} + (1-t)\mathbf{q}) > tH(\mathbf{p}) + (1-t)H(\mathbf{q})$$

Proof Applying Gibbs's inequality to $\mathbf{p}$ and $t\mathbf{p} + (1-t)\mathbf{q}$, yields

$$H(\mathbf{p}) \leq -\sum_r p_r \log(tp_r + (1-t)q_r)$$

Likewise

$$H(\mathbf{q}) \leq -\sum_r q_r \log(tp_r + (1-t)q_r)$$

with equality in both cases if and only if

$$p_r = tp_r + (1-t)q_r = q_r.$$

Forming

$$tH(\mathbf{p}) + (1-t)H(\mathbf{q}) \leq H(t\mathbf{p} + (1-t)\mathbf{q}),$$

by the above, yields the result; with strict inequality if $\mathbf{p} \neq \mathbf{q}$.
An amusing alternative proof is this:
Let X have distribution $\mathbf{p}$, and Y have distribution $\mathbf{q}$, and define the random variable M to be X with probability t or Y with probability $1-t$. Let Z be the indicator of the event that $M = X$. Then, in an obvious notation,

$$\begin{aligned} H(t\mathbf{p} + (1-t)\mathbf{q}) &= H(ZX + (1-Z)Y) \\ &= H(M) \geq H(M \mid Z) \\ &= tH(M \mid Z=1) + (1-t)H(M \mid Z=0) \\ &= tH(\mathbf{p}) + (1-t)H(\mathbf{q}) \end{aligned}$$

Finally, we note that the result follows directly from the fact that $-x \log x$ is concave. So

$$-(tp_r + (1-t)q_r)\log(tp_r + (1-t)p_r) \geq -tp_r \log p_r - (1-t)q_r \log q_r$$

in the above notation. Summing over r yields the result.
Equally importantly, we have this
Theorem. Let X and Y be the input and output of a channel

(a) Fixing the channel matrix $p(y \mid x)$, for all possible input distributions $\mathbf{p}$ of X, denote $I(X;Y)$ by $I(\mathbf{p})$. Then $I(\mathbf{p})$ is concave in $\mathbf{p}$.

(b) Fixing the distribution of X, for all possible channel matrices $p(y \mid x)$ denote $I(X;Y)$ by $I(p(y \mid x))$. Then $I(p(y \mid x))$ is convex in $p(y \mid x)$.

Proof (a). Let $\mathbf{p}$ and $\mathbf{q}$ be input distributions with corresponding output distributions $\mathbf{r}$ and $\mathbf{s}$. Then for $0 \leq t \leq 1$, using $I = H(Y) - H(Y \mid X)$,

$$\begin{aligned} & I(t\mathbf{p} + (1-t)\mathbf{q}) - tI(\mathbf{p}) - (1-t)I(\mathbf{q}) \\ &= H(t\mathbf{r} + (1-t)\mathbf{s}) - \sum_k (tp_k + (1-t)q_k) H(p(y \mid x_k)) \\ & -tH(\mathbf{r}) + t\sum_k p_k H(p(y \mid x_k)) - (1-t)H(\mathbf{s}) + (1-t)\sum_k q_k H(p(y \mid x_k)) \\ &= H(t\mathbf{r} + (1-t)\mathbf{s}) - tH(\mathbf{r}) - (1-t)H(\mathbf{s}) \geq 0 \end{aligned}$$

by the preceding theorem, as required. Alternatively we can prove this by showing that

$$\begin{aligned} & I(t\mathbf{p} + (1-t)\mathbf{q}) - tI(\mathbf{p}) - (1-t)I(\mathbf{q}) \\ &= t\sum_k r_k \log \frac{r_k}{p(y)} + (1-t)\sum_k s_k \log \frac{s_k}{p(y)} \end{aligned}$$

where $p(y)$ is the output distribution corresponding to the input distribution $t\mathbf{p} + (1-t)\mathbf{q}$. The result then follows by Gibbs's inequality.
And this result may also be proved by introducing an auxiliary indicator random variable,

Z, as we did above; this is left as an exercise.
More informally, write

$$I(X;Y) = H(Y) - \sum p_X(x) H(Y \mid X = x)$$

Here $H(Y)$ is concave in $\mathbf{p}$, by the preceding theorem, (because $p_Y(y)$ is linear in $\mathbf{p}$), and the second term is a linear function of $\mathbf{p}$. Hence I is concave in $\mathbf{p}$.
Proof (b). This is by similar methods, and is also left as an exercise for you.
We illustrate the use of this result by an example.
Example. Consider the channel with matrix

$$M = \begin{pmatrix} a & b & c \\ b & a & c \\ d & d & f \end{pmatrix}$$

For the input distribution (x, y, z), denote I by $I(x, y, z)$. Then by the apparent symmetry in M

$$I(x, y, z) = I(y, x, z)$$

But I is concave in the input distribution, so

$$I\left(\frac{x+y}{2}, \frac{x+y}{2}, z\right) \geq \frac{1}{2}(I(x, y, z) + I(y, x, z)) = I(x, y, z)$$

In seeking the capacity, we can therefore confine our search to input distributions of the form $(x, x, 1-2x)$. Writing $H_1 = H(a, b, c)$, and $H_2 = H(d, d, f)$, we have, (using $I = H(Y) - H(Y|X)$),

$$I = -2[(a+b)x + (1-2x)d] \log[(a+b)x + (1-2x)d]$$

$$-[2cx + (1-2x)f] \log[2cx + (1-2x)f] - 2xH_1 - (1-2x)H_2$$

The capacity is the maximum of this as x varies over $[0, \frac{1}{2}]$.

Note that if $f = c$ and $H_2 = H_1$, then the channel is useless. If $f = c$ and $H_2 > H_1$, then

$$I = H(d, d, f) - 2xH_1 - (1-2x)H_2$$

and $C = H_2 - H_1$, achieved when $x = \frac{1}{2}$.
More generally for $f \neq c$, differentiating to find a stationary value of I, we find that if

$$\min[c, f] \leq g = \left[1 + 2\frac{d}{f} 2^{D/(f-c)}\right]^{-1} \leq \max[c, f]$$

where D is the Kullback-Leibler divergence between (a, b, c) and (d, d, f), then there is a unique value of $x \in [0, \frac{1}{2}]$ that achieves the capacity. The remaining case arises when either $g < c < f$, or $g > c > f$, and then capacity is achieved by $x = \frac{1}{2}$.

Finally, note that while $H(\mathbf{p})$ is strictly concave in $\mathbf{p}$, $I(\mathbf{p})$ is not necessarily stricly concave in $\mathbf{p}$. To see this, consider the input and output, X and Y, of the channel with matrix

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix},$$

where capacity is achieved for any distribution $(1-a-b, a, b,)$ of X such that $a+b=\frac{1}{2}$.

3.5 Fano's inequality and the NCT converse

In §(1.8) we gave Shannon's sketch proof of the direct part of the noisy coding theorem, viz:-reliable transmission is precise at any rate up to the capacity C. In this section we prove a form of the converse, viz:-reliable transmission is not possible at any rate greater than C. To do this, we need the following result, which supplies a lower bound for the probability of error in using a channel, in terms of the equivocation $H(X \mid Y)$. As usual, the input to the channel is denoted by X.

Theorem. Fano's inequality
Let X and Y take values in the same alphabet A where $|A| = a$, and let E be the event that $X \neq Y$. Since Y is interpreted as our guess at the true value of X, on the basis of the output from the channel, $P(E)$ is called the probability of error. Then the Fano inequality is this:-

$$\begin{aligned} H(X \mid Y) &\leq H(P(E), 1-P(E)) + P(E)\log(a-1) \\ &\leq H(P(E), 1-P(E)) + P(E)\log a \\ &\leq 1 + P(E)\log a \end{aligned}$$

Proof
Define the indicator random variable J, which is 1 on E, or 0 on E^c. Then for each $y \in A$, J is a function of X and y. Hence

$$\begin{aligned} H(X \mid Y=y) &= H(X, J \mid Y=y) \\ &= H(J \mid Y=y) + H(X \mid J, Y=y) \\ &= H(J \mid Y=y) + P(E \mid Y=y)H(X \mid J=1, Y=y) + [1-P(E \mid Y=y)]H(X \mid J=0, Y=y) \\ &\leq H(J \mid Y=y) + P(E \mid Y=y)\log(a-1), \end{aligned}$$

since the entropy of X on the event $X=y$ is zero, and the entropy of X on the event $X \neq y$ is not greater than $\log(a-1)$. Multiplying by $P(Y=y)$, and summing over y, we have

$$\begin{aligned} H(X \mid Y) &\leq H(J \mid Y) + P(E)\log(a-1) \\ &\leq H(J) + P(E)\log(a-1) \end{aligned}$$

as required.

Corollary

If we seek to guess X without knowing Y, then the best guess is any value x such that $P(X = x) = p_{max}$ is maximal. Then $P(E) = 1 - p_{max}$ and Fano's inequality takes the form

$$H(X) \leq H(P(E), 1 - P(E)) + P(E) \log(a - 1),$$

a bound simply on the entropy $H(X)$ of X.

Alternatively, to show that $H(J \mid Y) \leq H(J)$, we may write that

$$H(J \mid Y = y) = H(P(E \mid Y = y),\ 1 - P(E \mid Y = y))$$

and then because entropy is concave in $\mathbf{p}$,

$$\sum_y P(Y = y) H(J \mid Y = y) \leq H(\sum_y P(Y = y)\ P(E \mid Y = y), 1 - \sum_y P(Y = y) P(E \mid Y = y))$$

$$= H(P(E),\ 1 - P(E))$$

An important application of Fano's inequality lies in the proof of a converse to the noisy coding theorem, thus:-

Theorem. For a channel having capacity C, no rate of transmission R is reliably achievable for $R > C$. More formally, there cannot exist a sequence of codebooks C_n, where C_n contains 2^{nR} codewords, such that the maximum probability of error for codewords in C_n converges to zero as $n \to \infty$.

Note: This is sometimes called the weak converse, because Wolfowitz's theorem asserts (much more strongly) that for rates $R > C$ the maximum probability of error converges geometrically fast to 1.

Proof. Suppose we have a sequence of codebooks C_n, where each C_n contains 2^{nR} codewords of length n, and $R > C$. If the maximum error goes to zero as $n \to \infty$, then so too must the average error. Therefore, we pick a codeword uniformly at random from the 2^{nR} available, and bound the average probability $\bar{e}$ of error using it. Denote the codeword by $\mathbf{X}$, the output by $\mathbf{Y}$, and the decoding by $\bar{\mathbf{X}} = g(\mathbf{Y})$. Then

$$\log 2^{nR} = nR = H(\mathbf{X})$$

, since $\mathbf{X}$ was selected uniformly

$$= H(\mathbf{X} \mid \bar{\mathbf{X}}) + I(\mathbf{X}; \bar{\mathbf{X}}),$$

by definition of I

$$\leq H(\mathbf{X} \mid \hat{\mathbf{X}}) + I(\mathbf{X}; \mathbf{Y}),$$

by the data-processing theorem, as $\bar{X} = g(\bar{Y})$

$$\leq 1 + P(\mathbf{X} \neq \bar{\mathbf{X}}) nR + I(\mathbf{X}; \mathbf{Y}),$$

by Fano's inequality

$$\leq 1 + P(\mathbf{X} \neq \bar{\mathbf{X}}) nR + nC$$

by the bound established for the capacity of the nth extension of a channel.
Hence,

$$\bar{e} = P(\hat{\mathbf{X}} \neq \mathbf{X}) \geq \frac{nR - nC - 1}{nR}$$
$$\rightarrow 1 - \frac{C}{R} \text{ as } n \rightarrow \infty$$
$$> 0, \text{ since } R > C.$$

Hence the maximum error cannot go to zero, and such rates are not achievable with any sequence of codes.

3.6 The noisy coding theorem for the BSC

We now give a rigorous proof of the noisy coding theorem for binary sources.[Which will do for any source, because it can be coded in binary with arbitrarily small loss of information, by Shannon's first (source) coding theorem.]
That is to say we prove this
Theorem. For any rate R, where $0 < R < C$, and C is the capacity of a binary symmetric channel, there exists a sequence of binary codes, with codebook sizes 2^{nR}, such that e_m, the maximum probability of error, goes to zero as $n \rightarrow \infty$. More strongly, in fact,

$$e_m < b2^{-an}, \text{ for some } a > 0, \ b > 0$$

Preparatory to the proof, we note these facts:
Codebook: The codebook comprises $2M$ binary strings of length n, which we denote by $\mathbf{c}_1, \ldots, \mathbf{c}_{2M}$. These codewords are obtained by making $2M$ independent selections uniformly at random from $B^n = \{0, 1\}^n$. [The reason for having a $2M$ codebook becomes apparent later on.]
Channel: The channel is binary symmetric, with error probability $p < \frac{1}{2}$, and capacity C, where

$$0 < C = 1 - H(p, 1-p) = 1 - H \leq 1$$

For any $0 < R < C$, we can find ϵ arbitrarily small such that $p + \epsilon < \frac{1}{2}$ and

$$R < 1 - H(p + \epsilon, 1 - p - \epsilon) < C \leq 1, \text{ as } H(p, 1-p) \text{ decreases on } \left[0, \frac{1}{2}\right).$$

Rate:The rate R of a codebook of size M, comprising strings of length n, is $R = \frac{1}{n} \log M$. We consider n so large that $2M \leq 2^n$ whenever $M = \lceil 2^{nR} \rceil$.
Decoder: The decoding function $g(\cdot)$ is the Hamming r-sphere decoder; that is, for any received n-string $\mathbf{y}$, (denoting the Hamming distance by $d(\mathbf{c}, \mathbf{y})$), we set

$$g(\mathbf{y}) = \mathbf{c}_k \text{ if } d(\mathbf{c}_k, \mathbf{y}) \leq r, \text{ and } \mathbf{c}_k \text{ is unique with this property;}$$

otherwise we declare an error, (which may be denoted by $\mathbf{c}_0$, say.) In what follows, we set

$$r = [n(p + \epsilon)] = \llcorner n(p + \epsilon) \lrcorner, \ \epsilon > 0$$

We shall need this
Lemma. Let the codeword $\mathbf{c}_k$ be sent and received as $\mathbf{y}$. Then

$$P(d(\mathbf{y}, \mathbf{c}_k) > r) \leq \exp[-\frac{1}{4} n\epsilon^2]$$

That is to say, the probability that at least $r+1$ errors are made by the channel among the n symbols of $\mathbf{c}_k$ is bounded by the RHS.

Proof Since the channel is discrete, and memoryless, and independent of our choice of codebook and codeword $\mathbf{c}_k$ to send, the number of errors is a binomial random variable V with parameters n and p. Hence, for any $t > 0$, setting $q = 1 - p$,

$$P(d(\mathbf{y}, \mathbf{c}_k) > r) = P(V > r)$$

$$\leq \sum_{k=r+1}^{n} \binom{n}{k} p^k q^{n-k} \exp[t(k - n(p+\epsilon))], \text{ as } n(p+\epsilon) \leq r < k,$$

$$\leq \sum_{k=0}^{n} \mathrm{e}^{-tn\epsilon} \binom{n}{k} [p\mathrm{e}^{t(1-p)}]^k [(1-p)\mathrm{e}^{-tp}]^{n-k}$$

$$= \mathrm{e}^{-tn\epsilon} (p\mathrm{e}^{tq} + q\mathrm{e}^{-tp})^n$$

$$\leq \mathrm{e}^{-tn\epsilon} (p\mathrm{e}^{t^2q^2} + q\mathrm{e}^{t^2p^2})^n, \text{ because } \mathrm{e}^x \leq x + \mathrm{e}^{x^2}, \text{ for all } x \in \mathbb{R}$$

$$\leq \mathrm{e}^{-tn\epsilon} \mathrm{e}^{t^2 n}$$

$$\leq \mathrm{e}^{-\frac{1}{4}n\epsilon^2} \text{ on identifying the minimum at } t = \frac{1}{2}\epsilon$$

Errors: The pointwise error of the code is

$$e_k = P(g(\mathbf{y}) \neq \mathbf{c}_k \mid \mathbf{c}_k \text{ was sent}),\ 1 \leq k \leq 2M$$

For this to occur, either $d(\mathbf{y}, \mathbf{c}_k) > r$, or $d(\mathbf{c}_j, \mathbf{y}) \leq r$ for some $j \neq k$. Since codewords were selected uniformly at random from $\{0,1\}^n$, the probability that $\mathbf{c}_j$ lies in the Hamming $\mathbf{r}$-sphere centre y is the volume of the r-sphere divided by 2^n, thus:

$$2^{-n} \sum_{k=0}^{r} \binom{n}{k} \leq 2^{nH(p+\epsilon, 1-p-\epsilon)-n}$$

Proof For any $0 < x < \frac{1}{2}$, note that $x/(1-x) < 1$ and $0 \leq [nx] \leq nx$. Hence

$$1 = (x + 1 - x)^n = \sum_{k=0}^{n} \binom{n}{k} x^k (1-x)^{n-k}$$

$$\geq \sum_{0}^{[nx]} \binom{n}{k} x^k (1-x)^{n-k} = (1-x)^n \sum_{0}^{[nx]} \binom{n}{k} \left(\frac{x}{1-x}\right)^k$$

$$\geq (1-x)^n \sum_{0}^{[nx]} \binom{n}{k} \left(\frac{x}{1-x}\right)^{nx}, \text{ as } \frac{x}{(1-x)} < 1$$

$$= \sum_{0}^{[nx]} \binom{n}{k} [x^x (1-x)^{1-x}]^n$$

$$= \sum_{0}^{[nx]} \binom{n}{k} 2^{-nH(x,1-x)}$$

Setting $x = p + \epsilon$, and recalling that $r = [n(p + \epsilon)]$ gives the required result.
Now suppose that U is uniformly distributed over $[1, \ldots, 2M]$, independently of the random choice of the codebook, and we send $\mathbf{c}U$. Then the probability of error is the average

$$\bar{e} = \frac{1}{2M} \sum_{k=1}^{2M} p(g(\mathbf{y}) \neq \mathbf{c}_k \mid \mathbf{c}_k \text{ is sent})$$

Naturally, it is required that $\bar{e}$ shall be small. More stringently, it will be required that the maximum pointwise error

$$e_m = \max_k P(g(\mathbf{y}) \neq \mathbf{c}_k \mid \mathbf{c}_k \text{is sent})$$

is also arbitrarily small. Remarkably, this can be readily shown if we first show that $\bar{e}$ is small, which we now do.
Note finally that this bound on the maximum probability of error gives the required bound on the probability of error for any distribution on the codewords that may be induced by the actual source.
Proof of the theorem
As described above, a codeword is selected at random to send, and denoted by X, and by construction the other codewords in the codebook are independent of this, and the resulting output Y. Hence

$$\begin{aligned}
\bar{e} &\leq P(d(X,Y) > r) + P(d(\mathbf{c}_k, Y) \leq r \text{ for some } \mathbf{c}_k \neq X) \\
&\leq \mathrm{e}^{-\frac{1}{4}n\epsilon^2} + (2M-1)P(d(\mathbf{c}_k, Y) \leq r) \\
&\leq \mathrm{e}^{-\frac{1}{4}n\epsilon^2} + 2\lceil 2^{nR} \rceil 2^{-n} 2^{nH(p+\epsilon,1-p-\epsilon)}, \text{ as shown above} \\
&\leq \mathrm{e}^{-\frac{1}{4}n\epsilon^2} + 2^{n(R-1+H(p+\epsilon,1-p-\epsilon)+2/n)}
\end{aligned}$$

But $R < 1 - H(p + \epsilon, 1 - p - \epsilon)$, so for some $a > 0$ and large enough n, $\bar{e} < 2^{-an}$.
Now we observe that there must be at least one $(n, 2M)$ codebook whose average error $\bar{e}_0 \leq \bar{e}$. But at least half the codewords in this codebook must have pointwise error $e_k \leq 2\bar{e}$, (for otherwise the overall average error of the codebook would exceed $\bar{e}$, which is a contradiction). So any M such codewords achieve the rate $R < C$ with maximum probability of error less than 2^{-an} as $n \to \infty$.
Corollary. Reliable transmission
A discrete memoryless source emits ω symbols per minute, and its output is encoded by Huffman's method for transmission through a binary symmetric channel with capacity C. Show that the rate T of transmission of the channel must satisfy $T > \omega H/C$, for reliable transmission, where H is the entropy of the source.
Proof Concatenate the sourcewords into blocks length b, with entropy bH. By Shannon's entropy bound this can be encoded into binary codewords of expected length $bH + 1$. So the expected rate of binary symbols presented to the channel is $A = \omega \left[\frac{bH+1}{b} \right]$. By the

noisy coding theorem, $e_{\max}$ is arbitrarily small if the rate A does not exceed TC symbols per minute. Hence

$$\omega\left[\frac{bH+1}{b}\right] < TC,$$

and letting $b \to \infty$ supplies the constraint $T > \omega H/C$.

Example: Feedback channel

Suppose that the usual channel of capacity C is augmented by noiseless feedback; that is to say the output Y is returned noiselessly (and instantly) to the sender which can either correct errors (or send more symbols). Show that the capacity of the augmented channel is still just C.

Solution Let the the capacity of the feedback channel be K, where $K \geq C$ trivially. As before, for $R > C$ there is an $(n, 2^{nR})$ codebook, and we obtain the average error probability by choosing a codeword to send uniformly at random. Call this n-string $\mathbf{W}$. The rth actual input to the channel, X_r is a function both of $\mathbf{W}$ and the transmitted signals $Y_1, \ldots, Y_{r-1}$ fed back.Then

$$I(\mathbf{W};\mathbf{Y}) = H(\mathbf{Y}) - H(\mathbf{Y} \mid \mathbf{W})$$

$$H(\mathbf{Y}) - \sum_1^n H(Y_r \mid Y_1, \ldots, Y_{r-1}, \mathbf{W}) \text{ by the chain rule}$$

$$= H(\mathbf{Y}) - \sum_1^n H(Y_r \mid Y_1, \ldots, Y_{r-1}, X_r, \mathbf{W})$$

because X_r is a function of $\mathbf{W}$ and the prior feedback. Hence

$$I(\mathbf{W};\mathbf{Y}) = H(\mathbf{Y}) - \sum_1^n H(Y_r \mid X_r)$$

because conditional on X_r, Y_r is independent of $\mathbf{W}$ and the past feedback. Thus

$$I(\mathbf{W};\mathbf{Y}) \leq \sum_1^n H(Y_r) - \sum_1^n H(Y_r \mid X_r)$$

$$= \sum_1^n I(X_r;Y_r) \leq nC$$

Now consider the probability of error, $\bar{e} = P(g(\mathbf{W}) \neq \mathbf{W})$, using Fano's inequality. Since $\mathbf{W}$ is uniform on 2^{nR} strings

$$nR = \log|\mathbf{W}| = H(\mathbf{W}) = H(\mathbf{W} \mid g(\mathbf{Y})) + I(\mathbf{W}; g(\mathbf{Y}))$$

$$\leq 1 + P(g(\mathbf{W}) \neq \mathbf{W})nR + I(\mathbf{W};\mathbf{Y}), \text{ by Fano's inequality}$$

where $g(\mathbf{Y})$ is the decoding, and the final term arises by use of the data-processing inequality. Hence

$$nR \leq 1 + \bar{e}nR + nC$$

by the above, so that

$$\bar{e} \geq \frac{n(R-C)-1}{nR} \to 1 - \frac{C}{R} > 0$$

as $n \to \infty$, since $R > C$.

3.7 Another interpretation of entropy and information

Shannon's information $I(X;Y)$ is interpreted in the context of noisy channels as the rate of transmission, and this interpretation is made significant by the noisy coding theorem in which the maximum of I, (under a suitable encoding), is the capacity C of the channel. It is of interest that the entropy H and $I(X;Y)$ are still significant in other contexts, in the absence of coding.
Here we consider elementary problems of long-term investment, or (what is theoretically and mathematically the same), compulsive gambling.

Example. Kelly betting.
Either because you are an investment manager, or because you are a pathological bettor, (or both), you must make the following bet independently each day:-
You stake some fraction $\phi X, (0 \leq \phi \leq 1)$, of your current fortune X on the flip of a coin. Thus, with probability $q = 1 - p$ you lose your stake, or, with probability p, your return is $a\phi X$. What should be your choice of ϕ?
First, note that if you were restricted to one bet, you maximize your expected gain by betting nothing if $ap < 1$, or betting your entire fortune X if $ap > 1$. But if you could play forever, and you choose $\phi = 1$, and you are bankrupted on the first play, then you have forfeited the chance of gains on further play. Second, note that if there is an optimal value of $\phi \in (0, 1)$, then it is the same at each stage.
Now, let your fortune on day n be $X_n, n \geq 0$.
Then

$$X_{n+1} = X_n M_{n+1}$$

where $M_1, M_2, \ldots$ are i.i.d.rvs, called the multipliers with distribution

$$P(M_n = 1 - \phi) = q = 1 - p$$

$$P(M_n = 1 - \phi + a\phi) = p$$

Then your fortune at the nth stage is

$$X_n = X_0 M_1 M_2 \ldots M_n$$

and

$$\log\left(\frac{X_n}{X_0}\right) = \sum_{r=1}^{n} \log M_r$$

where the random variables $\log M_r$ are i.i.d. with finite moments of all orders. Hence, by the law of large numbers (as used in the proof of the aymptotic equipartition property), we have

$$\frac{1}{n}\log\left(\frac{X_n}{X_0}\right) \to E \log M, \text{ as } n \to \infty$$

where the convergence proved is in probability. [But it also holds in any mean, and with probability 1.] More informally, we write this as

$$X_n \sim X_0 2^{nE \log M}$$

and note that your fortune grows exponentially, in the long run. It is natural to seek the maximum rate of long term growth, which is achieved by maximizing $r(\phi) = E \log M$, over choices of ϕ, where

$$r(\phi) = p \log(1 - \phi + a\phi) + (1 - p) \log(1 - \phi)$$

This policy is called the **Kelly criterion** (or strategy) and yields **Kelly betting**.
Note that $r(0) = 0$ and $r(1) = -\infty$. Furthermore, differentiating twice yields

$$r''(\phi) = \frac{-(a-1)^2 p}{(1-(a-1)\phi)^2} - \frac{(1-p)}{(1-\phi)^2} < 0 \text{ for all } \phi$$

so $r(\phi)$ is concave. Setting $r'(\phi) = 0$ yields a maximum, when $ap > 1$, at

$$\phi = \frac{ap - 1}{a - 1} \in (0, 1)$$

Otherwise, for $ap \leq 1$, the maximum of $r(\phi)$ is at $\phi = 0$. In the former case

$$r\left(\frac{ap-1}{a-1}\right) = \log a - (1-p)\log(a-1) - H(p, 1-p)$$

and the so-called **log-optimal return** on Kelly betting is given by

$$X_n \sim X_0 \left\{ \frac{a}{(a-1)^q} \right\}^n 2^{-nH},$$

where $H(p, 1-p)$ is the familiar entropy function.
Now, since $r(0) = 0$, $r(1) = -\infty$, and $r(\phi)$ is concave in $(0, 1)$, the function $r(\phi)$ has a unique zero in $[0, 1)$. That is to say, if you bet less than the Kelly fraction ϕ, your gain is still positive, but if you bet too far above the Kelly fraction your longterm rate of growth may be negative, even though $ap > 1$. The point at which $r(\phi) = 0$, where the switch occurs, is the root x of

$$(1-x)^q (1 + (a-1)x)^p = 1$$

If, for example, you are betting on a fair coin so that $p = q = \frac{1}{2}$, then $a > 2$ and

$$\phi = (\frac{1}{2}a - 1)/(a - 1)$$

and

$$x = (a - 2)/(a - 1)$$

So the interval (ϕ, x) from optimality to loss is of length

$$x - \phi = \frac{1}{2}(a-2)/(a-1)$$

This can be small even for advantageous ($ap > 1$) betting. For this reason, (among others), Kelly betting has been described as on the borderline between rational and insane investment policies. When seeking to use it, one should err on the side of caution, and bet low.

This example is somewhat artificial, as we were only allowed to bet on heads. In the real world we could also bet on tails. That is to say, one can bet on stocks rising or bet on them falling. We consider this more realistic problem, but, to avoid market technicalities, we consider a series of simple horse races.

Example. All-or-nothing market; alias bet-to-win
You may bet on an infinite sequence of i.i.d. horse races $X_1, X_2, \ldots$ as follows. There are m runners in each race, and the rth runner wins with probability $p_r = P(X_i = r)$. You may bet any fraction $b_r > 0$ of your fortune on the rth horse, $1 \le r \le m$, and $\sum_r b_r = 1$, If the rth runner wins then an amount equal to ω_r per unit stake on that horse is returned to you, and all your other stakes are lost. The vector $\mathbf{b}$ is thus your betting strategy for all races, and we have this:-
Theorem The rate of growth of the fortune Y_n, $n \ge 0$, of a gambler with strategy $\mathbf{b}$ is given by

$$Y_n \sim Y_0 2^{n\Delta(\mathbf{b},\mathbf{p})}$$

where $\Delta(\mathbf{b}, \mathbf{p}) = E \log b_X \omega_X$ is called the **doubling rate**. Furthermore

$$\Delta(\mathbf{b}, \mathbf{p}) \le \sum_r p_r \log \omega_r - H(\mathbf{p})$$

with equality if and only if $b_r = p_r$ for all $1 \le r \le m$, and $H(\mathbf{p})$ is the entropy of any race. This optimal strategy is called the Kelly proportional betting system.
Note that if the race has fair uniform odds, so that for $1 \le r \le m$ we have $\omega_r = m$, then the optimal doubling rate is

$$\Delta_{uf} = \log m - H(\mathbf{p})$$

which we recognize as the capacity of a symmetric channel with row vector $\mathbf{p}$. The fact that in this case

$$\Delta_{uf} + H(\mathbf{p}) = \log m$$

has been called the **conservation theorem**.
Proof. As for the coin, above, at each race

$$Y_{n+1} = Y_n b_{X_{n+1}} \omega_{X_{n+1}}, \text{ so that } \log \frac{Y_n}{Y_0} = \sum_1^n \log(b_{X_i} \omega_{X_i}).$$

Thus by the weak law of large numbers, as $n \to \infty$

$$\frac{1}{n} \log \left(\frac{Y_n}{Y_0} \right) \to E \log b_X \omega_X = \Delta(\mathbf{b}, \mathbf{p})$$

$$= \sum_r p_r \log \frac{b_r}{p_r} p_r \omega_r$$

$$= \sum p_r \log w_r + \sum p_r \log p_r - \sum p_r \log \frac{p_r}{b_r}$$

$$= \sum p_r \log \omega_r - H(\mathbf{p}) - d(\mathbf{p}, \mathbf{b})$$

$$\le \sum p_r \log \omega_r - H(\mathbf{p})$$

because the relative entropy $d(\mathbf{p}, \mathbf{b}) \geq 0$, with equality if and only if $p_r = b_r$ for all r. Note that for given pay-offs your doubling rate is smaller in a high-entropy race, as you might expect, as there is more uncertainty about the result.
Note also that if $\sum_r \frac{1}{\omega_r} = 1$, then the pay-offs are said to be fair. [This nomenclature arises because you have the available betting strategy

$$b_r = \frac{1}{\omega_r}$$

, so that the expected return on a unit stake is exactly

$$\sum b_r \omega_r p_r = \sum \frac{1}{\omega_r} \omega_r p_r = 1$$

which is fair. But this is not necessarily the log-optimal long term growth strategy.]
In the fair case, setting $\omega_r^{-1} = s_r$

$$\Delta(\mathbf{b}, \mathbf{p}) = \sum p_r \log \left(\frac{b_r}{p_r} \frac{p_r}{\omega_r^{-1}} \right)$$

$$= d(\mathbf{p}, \mathbf{s}) - d(\mathbf{p}, \mathbf{b})$$

where $d(\cdot, \cdot)$ is the relative entropy function, as usual.
The distribution $s_r = \omega_r^{-1}$ can be seen as the bookies estimate of the true odds p_r, while b_r can be seen as yours. You win, i.e. $\Delta > 0$, if your betting strategy $\mathbf{b}$ is closer to $\mathbf{p}$ than the bookies fair odds distribution $\mathbf{s}$.
Almost always, in real races, $\sum_r \frac{1}{\omega_r} > 1$, which is called subfair. The optimal strategy is not to bet your entire fortune, but (as in the case of the coin) to retain part at each stage. Very rarely, it may happen that $\sum_r \frac{1}{\omega_r} < 1$. This is called superfair, and by a famous theorem there then exists a betting strategy $\mathbf{b}$ that guarantees a risk-free profit (called an arbitrage). Once found, your entire fortune can be staked for a sure win. Note that this is nevertheless not (necessarily) the log-optimal Kelly strategy.
In this particular superfair case, you may set

$$b_r = \frac{1}{\omega_r} (\sum \omega_r^{-1})^{-1}$$

as your betting strategy; then, whichever horse wins, your return is

$$X(\sum \omega_r^{-1})^{-1} > X$$

Both in real life and in horse races, your strategy is influenced by side-information, e.g. the horse's mouth. [This may, in other contexts, be described as insider trading.] Denote this by the r.v. Y, so that X and Y are jointly distributed as $p(r, y)$, and your betting strategy, given $Y = y$ now $b(r \mid y)$. Then the doubling rate is, (in a slightly different notation),

$$\Delta_Y = \Delta(b(r \mid y), p(r, y))$$

$$= \sum_r p(r, y) \, log(b(r \mid y) \omega_r)$$

$$= \sum_r p(r,y) \log \left(\frac{b(r \mid y)}{p(r \mid y)} p(r \mid y) \omega_r \right)$$

$$= \sum_r p(r,y) \log \omega_r - d(p(r \mid y), b(r \mid y)) - H(X \mid Y)$$

$$\leq \sum_r p_r \log \omega_r - H(X \mid Y), \text{ with equality iff } b(r \mid y) = p(r \mid y)$$

Hence the difference in the doubling rate, (between knowing Y or not knowing it) is

$$\Delta_Y - \Delta = H(X) - H(X \mid Y) = I(X;Y)$$

the mutual information between the race and your side knowledge of the race. If $I = 0$, the side information was irrelevant.