Introduction

We model communication:

$$\underbrace{\text{SOURCE}}_{\text{message}} \to \underbrace{\text{ENCODER}}_{\text{codewords}} \xrightarrow[\text{errors,noise}]{\text{CHANNEL}} \underbrace{\text{DECODER}}_{\text{recieved word error correction}} \to \underbrace{\text{RECIEVER}}_{\text{message}}.$$

Examples: optical signals, electrical telegraph, SMS (compression), postcodes, CDs (error correction), zip/gz files (compression).
Given a source and a channel, modelled probabilistically, the basic problem is to design an encoder and decoder to transmit messages economically (noiseless coding; compression) and reliably (noisy coding).

Examples:

- Noiseless coding: Morse code: common letters are assigned shorter codewords, e.g $A \mapsto \bullet -$, $E \mapsto \bullet$, $Q \mapsto - - \bullet -$, $S \mapsto \bullet\bullet\bullet$, $O \mapsto - - -$, $Z \mapsto - - \bullet\bullet$. Noiseless coding is adapted to source.
- Noisy coding: Every book has an ISBN $a_1, a_2, \dots, a_9, a_{10}$, $a_i \in \{0, 1, \dots, 9\}$ for $1 \leq i \leq 9$ and $a_{10} \in \{0, 1, \dots 9, X\}$ with $\sum_{j=1}^{10} ja_j \equiv 0 \pmod{11}$. This detects common errors - e.g one incorrect digit, transposition of two digits. Noisy coding is adapted to the channel.

Plan:

(I) Noiseless coding - entropy

(II) Error correcting codes - noisy channels

(III) Information theory - Shannon's theorems

(IV) Examples of codes

(V) Cryptography

Books: [GP], [W], [CT], [TW], Buchmann, Körner. Online notes: Carne, Körner.

Basic Definitions

Definition (Communication channel)**.** A *communication channel* accepts symbols from a alphabet $\mathcal{A} = \{a_1, \dots, a_r\}$ and it outputs symbols from alphabet $\mathcal{B} = \{b_1, \dots, b_s\}$. Channel modelled by the probabilities $\mathbb{P}(y_1 \dots y_n \text{ recieved} | x_1 \dots x_n \text{sent})$.
A *discrete memoryless channel* (DMC) is a channel with

$$p_{ij} = \mathbb{P}(b_j \text{ recieved} | a_i \text{ sent})$$

the same for each channel use and independent of all past and future uses. The channel matrix is $P = (b_{ij})$, a $r \times s$ stochastic matrix.

Definition (Binary symmetric channel)**.** The *binary symmetric channel* (BSC) with error probability $p \in [0, 1)$ from $\mathcal{A} = \mathcal{B} = \{0, 1\}$. The channel matrix is

$$\begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}.$$

A symbol is transmitted correctly with probability $1 - p$. Usually assume $p < 1/2$.

The *binary erasure channel* (BEC) has $\mathcal{A} = \{0, 1\}$, $\mathcal{B} = \{0, 1, *\}$. The channel matrix is

$$\begin{pmatrix} 1-p & 0 & p \\ 0 & 1-p & p \end{pmatrix}.$$

So $p = \mathbb{P}(\text{symbol can't be read})$.

Definition. We model n uses of a channel by the nth extension, with input alphabet $\mathcal{A}^n$ and output alphabet $\mathcal{B}^n$. A *code C of length n* is a function $\mathcal{M} \to \mathcal{A}^n$ where $\mathcal{M}$ is the set of possible messages. Implicitly we also have a decoding rule $\mathcal{B}^n \to \mathcal{M}$. The *size* of C is $m = |\mathcal{M}|$. The *information rate* is $\rho(C) = \frac{1}{n} \log_2 m$. The *error rate* is $\hat{e}(C) = \max_{x \in \mathcal{M}} \mathbb{P}(\text{error}|x \text{ sent})$.

Remark. For the remainder of the course we write log instead of $\log_2$.

Definition. A channel can *transmit reliably at rate R* if there exists $(C_n)_{n=1}^{\infty}$ with each C_n a code of length n such that

$$\lim_{n\to\infty} \rho(C_n) = R \ \& \ \lim_{n\to\infty} \hat{e}(C_n) = 0.$$

The *capacity* is the supremum of all reliable transmission rates. We'll see in Chapter 9 that a BSC with error probability $p < 1/2$ has non-zero capacity.

1 Noiseless coding

1.1 Prefix-free codes

For an alphabet $\mathcal{A}$, $|\mathcal{A}| < \infty$, let $\mathcal{A}^* = \bigcup_{n\geq 0} \mathcal{A}^n$, the set of all finite strings from $\mathcal{A}$. The *concatenation* of strings $x = x_1 \dots x_r$ and $y = y_1 \dots y_s$ is $xy = x_1 \dots x_r y_1 \dots y_s$.

Definition. Let $\mathcal{A}, \mathcal{B}$ be alphabets. A code is a function $c : \mathcal{A} \to \mathcal{B}^*$. The strings $c(a)$ for $a \in \mathcal{A}$ are called *codewords* or *words* (CWS).

Example 1.1 (Greek fire code)**.** $\mathcal{A} = \{\alpha, \beta, \dots, \omega\}$ (greek alphabet), $\mathcal{B} = \{1, 2, 3, 4, 5\}$., $c : \alpha \mapsto 11, \beta \mapsto 12, \dots, \psi \mapsto 53$, $\omega \mapsto 54$. xy means hold up x torches and another y torches nearby.

Example 1.2. $\mathcal{A}$ = words in a dictionary, $\mathcal{B} = \{A, B, \dots, Z, \omega\}$. $c : \mathcal{A} \to \mathcal{B}$ splits the word and follows with a space. Send message $x_1 \dots x_n \in \mathcal{A}^*$ as $c(x_1) \dots c(x_n) \in \mathcal{B}^*$. So c extends to a function $c^* : \mathcal{A}^* \to \mathcal{B}^*$.

Definition. c is said to be *decipherable* if the induced map c^* (as in the previous example) is injective. In other words, each string from $\mathcal{B}$ corresponds to at most one message.

Clearly if c is decipherable, it is necessary for c to be injective. However it is not sufficient:

Example 1.3. $\mathcal{A} = \{1, 2, 3, 4\}$, $\mathcal{B} = \{0, 1\}$. Define $c : 1 \mapsto 0$, $2 \mapsto 1$, $3 \mapsto 00$, $4 \mapsto 01$. Then $c^*(114) = 0001 = c^*(312) = c^*(144)$ yet c is injective.

Notation: $|\mathcal{A}| = m$, $|\mathcal{B}| = a$, call c am a-ary code of size m. For example a 2-ary code is a binary one, and a 3-ary code is a ternary code.

Our aim is to construct decipherable codes with short word lengths. Assuming c is injective, the following codes are always decipherable:

(i) A block code has all codewords of the same length (e.g Greek fire code);

(ii) A comma code reserves a letter from $\mathcal{B}$ to signal the end of a word (e.g Example 1.2);

(iii) A prefix-free code is a code where no codeword is a prefix of any other distinct word (if $x, y \in \mathcal{B}^*$ then x is a prefix of y if $y = xz$ for some string $z \in \mathcal{B}^*$).

(i) and (ii) are special cases of (iii). As we can decode the message as it is recieved, prefix-free codes are sometimes called *instantaneous*.

Exercise: find a decipherable code which is not prefix-free.

Definition (Kraft's inequality)**.** $|\mathcal{A}| = m$, $|\mathcal{B}| = a$, $c : \mathcal{A} \to \mathcal{B}^*$ has word lengths $l_1, \dots, l_m$. Then Kraft's inequality is

$$\sum_{i=1}^{m} a^{-l_i} \leq 1. \tag{$*$}$$

Theorem 1.1. *A prefix-free code exists if and only if Kraft's inequality* $(*)$ *holds.*

Proof. Rewrite $(*)$ as

$$\sum_{l=1}^{s} n_l a^{-l} \leq 1, \tag{$**$}$$

where n_l is the number of codewords with length l, and $s = \max_{1 \le i \le m} l_i$.

Now if $c : \mathcal{A} \to \mathcal{B}^*$ is prefix-free,

$$n_1 a^{s-1} + n_2 a^{s-2} + \ldots + n_{s-1} a + n_a \le a^s.$$

Indeed the LHS is the number of strings of length s in B with some codeword of c as a prefix, and the RHS is the total number of strings of length S. Dividing through by a^s we get $(**)$.

Now given $n_1, \ldots, n_s$ satisfying $(**)$, we try to construct a prefix-free code c with n_l codewords of length l, $\forall l \le s$. Proceed by induction on s, $s = 1$ is clear (since $(**)$ gives $n_1 \le a$ so can construct code).

By the induction hypothesis, there exists a prefix-code $\hat{c}$ with n_l codewords of length l for all $l \le s - 1$. Then $(**)$ implies

$$n_1 a^{s-1} + n_2 a^{s-2} + \ldots n_{s-1} a + n_s \le a^s.$$

The first $s-1$ terms on the LHS sum to the number of strings of length s with a codeword of $\hat{c}$ as a prefix and the RHS is the number of strings of length s. Hence we can add at least n_s new codewords of length s to $\hat{c}$ and maintain the prefix-free property.

$\square$

Remark. This proof is constructive: just choose codewords in order of increasing length, ensuring that no previous codeword is a prefix.

Theorem 1.2 (McMillan)**.** *Any decipherable code satisfies Kraft's inequality.*

Proof (Karush, 1961). Let $c : \mathcal{A} \to \mathcal{B}^*$ be a decipherable code with word lengths $l_1, \ldots, l_m$. Set $s = \max_{1 \le i \le m} l_i$. For $R \in \mathbb{N}$

$$\left(\sum_{i=1}^{m} a^{-l_i} \right)^R = \sum_{l=1}^{Rs} b_l a^{-l}, \tag{$\dagger$}$$

where b_l is the number of ways of choosing R codewords of total length l. Since c is decipherable, any string of length l formed from codewords must correspond to at most one sequence of codewords, i.e $b_l \le |\mathcal{B}^l| = a^l$. Subbing this into $(\dagger)$

$$\left(\sum_{i=1}^{m} a^{-l_i} \right)^R \le \sum_{i=1}^{Rs} a^l a^{-l} = Rs,$$

so

$$\sum_{i=1}^{m} a^{-l_i} \le (Rs)^{1/R} \to 1 \text{ as } R \to \infty.$$

Hence $\sum_{i=1}^{m} a^{-l_i} \le 1$. $\square$

Corollary 1.3. *A decipherable code with prescribed word lengths exists if and only if a prefix-free code with the same word lengths exists.*

Proof. Combine previous two theorems. $\square$

Therefore we can restrict our attention to prefix-free codes.

2 Shannon's Noiseless Coding Theorem

Entropy is a measure of 'randomness' or 'uncertainty'. Suppose we have a random variable X taking a finite set of values $x_1, \dots, x_n$ with probabilities $p_1, \dots, p_n$ respectively. The *entropy* $H(X)$ of X is the expected number of fair coin tosses needed to simulate X (roughly speaking).

Example 2.1. Suppose $p_1 = p_2 = p_3 = p_4 = 1/4$. Identify (x_1, x_2, x_3, x_4) with (HH, HT, TH, TT). Then the entropy is 2.

Example 2.2. Suppose $(p_1, p_2, p_3, p_4) = (1/2, 1/4, 1/8, 1/8)$. Identify (x_1, x_2, x_3, x_4) with (H, TH, TTH, TTT). Then the entropy is

$$\frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{8} \times 3 = \frac{7}{4}.$$

In a sense, the previous example (2.1) was 'more random' than this.

Definition (Entropy)**.** The *entropy* of X is

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

(Recall that $\log =: \log_2$ here.) Note $H(X) \geq 0$. It is measured in *bits* (binary digits). Conventionally, we take $0 \log 0 = 0$.

Example 2.3. Take a biased coin $\mathbb{P}(H) = p$, $\mathbb{P}(T) = 1-p$. Write $H(p, 1-p) := H(p)$. Then

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

Note that $H'(p) = \log \frac{1-p}{p}$. Hence the entropy is maximised for $p = 1/2$ (giving entropy 1).

Proposition 2.1 (Gibbs' inequality)**.** *Let $(p_1, \dots, p_n), (q_1, \dots, q_n)$ be probability distributions. Then*

$$-\sum_{i=1}^{n} p_i \log p_i \leq -\sum_{i=1}^{n} p_i \log q_i.$$

(The RHS is sometimes called the cross entropy or mixed entropy) Furthermore we have equality iff $p_i = q_i$ for all i.

Proof. Since $\log x = \frac{\ln x}{\ln 2}$, we may replace log with ln. Put $I = \{1 \leq i \leq n : p_i \neq 0\}$. Now $\ln x = x - 1$ for all $x > 0$ with equality iff $x = 1$. Hence $\ln \frac{q_i}{p_i} \leq \frac{q_i}{p_i} - 1$ for all $i \in I$. So

$$\sum_{i \in I} p_i \ln \frac{q_i}{p_i} \leq \underbrace{\sum_{i \in I} q_i}_{\leq 1} - \underbrace{\sum_{i \in I} p_i}_{=1} \leq 0$$

$$\implies -\sum_{i \in I} p_i \ln p_i \leq -\sum_{i \in I} p_i \ln q_i$$

$$\implies -\sum_{i=1}^{n} p_i \ln p_i \leq -\sum_{i=1}^{n} p_i \ln q_i.$$

If equality holds, then $\sum_{i\in I} q_i = 1$ and $\frac{p_i}{q_i} = 1$ for all $i \in I$. So $q_i = p_i$ for all $1 \leq i \leq n$. □

Corollary 2.2. *$H(p_1, p_2, \ldots, p_n) \leq \log n$ with equality iff $p_1 = p_2 = \ldots = p_n = 1/n$.*

Proof. Take $q_1 = q_2 = \ldots = q_n = 1/n$ in Gibbs' inequality. □

Let $\mathcal{A} = \{\mu_1, \ldots, \mu_m\}$, $|\mathcal{B}| = a$ $(m, n \geq 2)$. The random variable X takes values $\mu_1, \ldots, \mu_m$ with probabilities $p_1, \ldots, p_m$.

Definition. If $c : \mathcal{A} \to \mathcal{B}^*$ is a code, we say it is *optimal* if has the smallest possible expected word length. i.e $\mathbb{E}S := \sum_{i=1}^{n} p_i l_i$ is minimal amongst all decipherable codes.

Theorem 2.3 (Shannon's Noiseless Coding Theorem)**.** *The expected word length $\mathbb{E}S$ of an optimal code satisfies*

$$\frac{H(X)}{\log a} \leq \mathbb{E}S < \frac{H(X)}{\log a} + 1.$$

Remark. The lower bound is actually true for any decipherable code.

Proof. We first get the lower bound. Let $c : \mathcal{A} \to \mathcal{B}^*$ be decipherable with word lengths $l_1, \ldots, l_m$. Let $q_i = \frac{a^{-l_i}}{D}$ where $D = \sum_{i=1}^{m} a^{-l_i}$. Note $\sum_{i=1}^{m} q_i = 1$. By Gibbs' inequality

$$\begin{aligned} H(X) &\leq -\sum_{i=1}^{m} p_i \log q_i \\ &= -\sum_{i=1}^{m} p_i(-l_i \log a - \log D) \\ &= \left(\sum_{i=1}^{m} p_i l_i\right) \log a + \log D. \end{aligned}$$

By McMillan, $D \leq 1$ so $\log D \leq 0$. Hence

$$H(X) \leq \left(\sum_{i=1}^{m} p_i l_i\right) \log a \implies \frac{H(X)}{\log a} \leq \mathbb{E}S.$$

And we have equality iff $p_i = a^{-l_i}$ for some integers $l_1, \ldots, l_m$. Note we have only used decipherability so far.

Now we get the upper bound. Take $l_i = \lceil -\log_a p_i \rceil$. Then

$$-\log_a p_i \leq l_i < -\log_a p_i + 1.$$

Hence $\log_a p_i \geq -l_i$, so $p_i \geq a^{-l_i}$. Therefore $\sum_{i=1}^m a^{-l_i} \leq \sum_{i=1}^m p_i = 1$. By Kraft's inequality, there exists a prefix-free code c with word lengths $l_1, \ldots, l_m$. c has expected word length

$$\mathbb{E}S = \sum_{i=1}^m p_i l_i < \sum_{i=1}^m p_i(-\log_a p_i + 1) = \frac{H(X)}{\log a} + 1.$$

$\square$

Example 2.4 (Shannon-Fano Coding)**.** We mimic the above proof: given $p_1, \ldots, p_m$, set $l_i = \lceil -\log_a p_i \rceil$. Construct a prefix-free code with word lengths l_i by choosing codewords in order of increasing length, ensuring any new codeword has no previous codeword as a prefix (Kraft's inequality ensures we can do this).

Example 2.5. Take $a = 2, m = 5$.

i	p_i	$\lceil -\log_2 p_i \rceil$	code
1	0.4	2	00
2	0.2	3	010
3	0.2	3	011
4	0.1	4	1000
5	0.1	4	1001

Then $\mathbb{E}S = \sum_{i=1}^m p_i l_i = 2.8$, $H = H/\log a = 2.12$. [See also Carne p13.]

3 Huffman Coding

How to construct an optimal code? Take $\mathcal{A} = \{\mu_1, \dots, \mu_m\}$, $p_i = \mathbb{P}(X = \mu_i)$. For simplicitly take $|\mathcal{B}| = a = 2$. Without loss of generality $p_1 \geq p_2 \geq \dots \geq p_m$. Huffman gave an inductive definition of codes that we can prove are optimal. If $m = 2$, we take codewords $0, 1$. If $m > 2$, first take the Huffman code for messages $\mu_1, \dots, \mu_{m-2}, \nu$ with probabilities $p_1, \dots, p_{m-2}, p_{m-1} + p_m$. Then append 0 (respectively 1) to the codeword for ν to give a codeword for μ_{m-1} (respectively μ_m).

Notes.

- Huffman codes are prefix-free;
- Huffman codes are not unique: choice is needed if some of the p_i are equal.

Example 3.1. Revisit Example 2.5. We have

i	p_i	$c^{(1)}$	$p_i^{(2)}$	$c^{(2)}$	$p_i^{(3)}$	$c^{(3)}$	$p_i^{(4)}$	$c^{(4)}$
1	0.4	1	0.4	1	0.4	1	0.6	0
2	0.2	01	0.2	01	0.4	00	0.4	1
3	0.2	000	0.2	000	0.2	01		
4	0.1	0010	0.2	001				
5	0.1	0011						

Theorem 3.1. *Huffman codes are optimal (Huffman, 1952).*

Proof. We show by induction on m that Huffman codes of size $m = |\mathcal{A}|$ are optimal.

$\underline{m = 2}$: codewords are $0, 1$ - clearly optimal.

$\underline{m > 2}$: let c_m be a Huffman code for X_m, which takes values $\mu_1, \dots, \mu_m$ with probabilities $p_1 \geq p_2 \geq \dots \geq p_m$; each c_m is constructed from Huffman code c_{m-1} for X_{m-1} which takes values $\mu_1, \dots, \mu_{m-2}, \nu$ with probabilities $p_1, \dots, p_{m-2}, p_{m-1} + p_m$. Then the expected word length is

$$\mathbb{E}S_m = \mathbb{E}S_{m-1} + p_{m-1} + p_m. \tag{$*$}$$

Let c'_m be an optimal code for X_m. Wlog c'_m is still prefix-free. Wlog the last two codewords of c'_m have maximal length and differ only in the final position (see next lemma). Say

$$c'_m(\mu_{m-1}) = y0,\ c'_m(\mu_m) = y1 \text{ for some } y \in \{0,1\}^*.$$

Let c'_{m-1} be some prefix-free code for X_{m-1}, given by

$$c'_{m-1}(\mu_i) = \begin{cases} c'_m(\mu_i) & 1 \leq i \leq m-2 \\ c'_{m-1}(\nu) = y & \end{cases}.$$

Then the expected word length satisfies

$$\mathbb{E}S'_m = \mathbb{E}S'_{m-1} + p_{m-1} + p_m. \tag{$**$}$$

By the inductive hypothesis, c_{m-1} is optimal, so $\mathbb{E}S_{m-1} \leq \mathbb{E}S'_{m-1}$. By $(*)$ and $(**)$ this implies $\mathbb{E}S_m \leq \mathbb{E}S'_m$.

□

Lemma 3.2. *Suppose letters $\mu_1, \dots, \mu_m$ in $\mathcal{A}$ are sent with probabilities $p_1, p_2, \dots, p_m$. Let c be an optimal (prefix-free) code with word lengths $l_1, \dots, l_m$. Then*

(i) If $p_i > p + j$, then $l_i \leq l_j$;

(ii) Amongst all codewords of maximal length there exist two that differ only in the final digit.

Proof. (i) is obvious. For (ii), could otherwise just delete the final digit of the codeword of maximal length (since prefix-free). □

Remark. Note not all optimal codes are Huffman (look at the case $m = 4$).

Our main result says that if we have a prefix-free optimal code with word lengths $l_1, \dots, l_m$ and associated probabilities $p_1, \dots, p_m$, then there is a Huffman code with these word lengths.

4 Joint Entropy

If X, Y are random variables with values in $\mathcal{A}$ and $\mathcal{B}$ respectively, then (X, Y) is a random variable with values in $\mathcal{A} \times \mathcal{B}$, and the *entropy* $H(X, Y)$ is called the joint entropy, given by

$$H(X, Y) = -\sum_{x \in \mathcal{A}} \sum_{y \in \mathcal{B}} \mathbb{P}(X = x, Y = y) \log \mathbb{P}(X = x, Y = y).$$

This generalises to any finite number of random variables.

Lemma 4.1. *Let X, Y be random variables taking values in $\mathcal{A}$ and $\mathcal{B}$ respectively. Then*

$$H(X, Y) \leq H(X) + H(Y),$$

with equality if and only if X and Y are independent.

Proof. Write $\mathcal{A} = \{x_1, \dots, x_m\}$, $\mathcal{B} = \{y_1, \dots, y_n\}$. Let

$$p_{ij} = \mathbb{P}(X = x_i, Y = Y_j),\ p_i = \mathbb{P}(X = x_i),\ q_j = \mathbb{P}(Y = y_j).$$

Apply Gibbs' inequality to the probability distributions $\{p_{ij}\}$ and $\{p_i q_j\}$ to obtain

$$\begin{aligned} -\sum_{i,j} p_{ij} \log p_{ij} &\leq -\sum_{i,j} p_{ij} \log(p_i q_j) \\ &= -\sum_i \left(\sum_j p_{ij} \right) \log p_i - \sum_j \left(\sum_i p_{ij} \right) \log q_j \\ &= -\sum_i p_i \log p_i - \sum_j q_j \log q_j \\ &= H(X) + H(Y). \end{aligned}$$

With equality if and only if $p_{ij} = p_i q_j$ for all i, j. $\square$

Error-correcting codes

5 Noisy channels and Hamming's code

Definition. A *binary* $[m, n]$*-code* is a subset C of $\{0,1\}^n$ of size $m = |C|$. n is the length of the code and the elements of C are called codewords.

We use an $[n, m]$-code to send one of m messages through a BSC (binary symmetric channel) making n uses of the channel. Clearly $1 \leq m \leq 2^n$, so $0 \leq \frac{1}{n}\log m \leq 1$.

Definition. For any $x, y \in \{0,1\}^n$ the *Hamming distance* is

$$d(x,y) = |\{i : 1 \leq i \leq n, x_i \neq y_i\}|.$$

Definition.

(i) The *ideal observer* decoding rule decodes $x \in \{0,1\}^n$ as $c \in C$ maximising $\mathbb{P}(c \text{ sent}|x \text{ recieved})$.

(ii) The *maximum likelihood* decoding rule decodes $x \in \{0,1\}^n$ as $c \in C$ maximising $\mathbb{P}(x \text{ recieved}|c \text{ sent})$

(iii) The *minimum distance* decoding rule decodes $x \in \{0,1\}$ as $c \in C$ minimising $d(x, C)$.

Lemma 5.1.

(a) If all the messages are equally likely, then (i) and (ii) above are equivalent.

(b) If $p < 1/2$ (error probability) then (ii) and (iii) are equivalent.

Remark. If $p = 1/2$ the code is called *useless*. If $p = 0$ the code is called *lossless*.

Proof.

(a) We have

$$\mathbb{P}(c \text{ sent}|x \text{ recieved}) = \frac{\mathbb{P}(c \text{ sent}, x \text{ recieved})}{\mathbb{P}(x \text{ recieved})} = \frac{\mathbb{P}(c \text{ sent})}{\mathbb{P}(x \text{ recieved})}\mathbb{P}(x\text{recieved}|c\text{sent}).$$

So by hypothesis, $\mathbb{P}(c \text{ sent})$ is independent of $c \in C$. So for fixed x, maximising $\mathbb{P}(c \text{ sent}|x \text{ recieved})$ is the same as maximising $\mathbb{P}(x \text{ recieved}|c \text{ sent})$.

(b) Let $r = d(x, c)$. Then $\mathbb{P}(x \text{ recieved}|c \text{ sent}) = p^r(1-p)^{n-r} = (1-p)^n \left(\frac{p}{1-p}\right)^r$. Since $p < 1/2$. $\frac{p}{1-p} < 1$. So maximising $\mathbb{P}(x \text{ recieved}|c \text{ sent})$ is the same as minimising r.

$\square$

We choose to use minimum distance decoding from now on.

Example 5.1. Suppose 000, 111 are sent with probabilities $\alpha = 9/10$, $\beta = 1/10$ respectively through a BSC with error probability $p = 1/4$. Suppose 110 is recieved. Then

$$\mathbb{P}(000 \text{ sent}|110 \text{ recieved}) = \frac{\alpha p^2(1-p)}{\alpha p^2(1-p) + (1-\alpha)p(1-p)^2} = \frac{3}{4},$$

$$\text{similarly } \mathbb{P}(111 \text{ sent}|110 \text{ recieved}) = \frac{1}{4}.$$

So the ideal observer decodes as 000. But the maximum likelihood/minimum distance rules decode as 111.

Remarks.

- Minimum distance decoding may be expensive in terms of time and storage if $|C|$ is large.
- Need to specify a convention in case there is no unique maximiser (e.g make a random choice, or request the message is sent again).

We aim to detect, or even correct errors.

Definition. A code C is

- *d-error* detecting if changing up to d digits in each codeword can never produce another codeword. In other words, each codeword is of Hamming distance greater than d from every other codeword.
- *e-error* correcting if knowing that $x \in \{0,1\}^n$ differs fom a codeword in at most e places we can deduce the codeword.

Examples.

(a) A *repitition code* of length n has codewords $\underbrace{00\ldots0}_{n \text{ times}}, \underbrace{11\ldots1}_{n \text{ times}}$. This is a $[n,2]$-code. It is $(n-1)$-error detecting and $\lfloor \frac{n-1}{2} \rfloor$-error correcting. But the information rate is only $1/n$.

(b) A *simple parity check code* or *paper tape code*: identify $\{0,1\}$ with $\mathbb{F}_2$ and let $C = \{(x_1,\ldots,x_n) \in \{0,1\}^n : \sum_{i=1}^n x_i = 0\}$. This is a $[n, 2^{n-1}]$-code, 1-error detecting but cannot correct errors. The information rate is $\frac{n-1}{n}$.

(c) Hamming's original code (1950): a 1-error correcting binary $[7,16]$-code. Take $C \subseteq \mathbb{F}_2^7$ where

$$C = \{c \in \mathbb{F}_2^7 : c_1+c_3+c_5+c_7 = 0, c_2+c_3+c_6+c_7 = 0, c_4+c_5+c_6+c_7 = 0\}.$$

The bits c_3, c_5, c_6, c_7 are arbitrary and c_1, c_2, c_4 are forced (called the check digits) so $|C| = 2^4$. To decode: suppose we recieve $x \in \mathbb{F}_2^7$. We form the *syndrome*: $z = z_x = (z_1, z_2, z_4) \in \mathbb{F}_2^7$ where

$$\begin{aligned} z_1 &= x_1 + x_3 + x_5 + x_7 \\ z_2 &= x_2 + x_3 + x_6 + x_7 \\ z_4 &= x_4 + x_5 + x_6 + x_7. \end{aligned}$$

If $x \in C$, then $z_x = (0, 0, 0)$. If $d(x, c) = 1$ for some $c \in C$, then place where x and c differ is given by $z_1 + 2z_2 + 4z_4$ (not mod 2). Check: if $x = c + e_i$ where e_i has all 0's except a 1 in the ith position, then $z_x = z_{e_i}$, so check for each $1 \le i \le 7$.

Lemma 5.2. *The Hamming distance is a metric on* $\mathbb{F}_2^n$.

Proof. Trivial. □

Definition. The *minimum distance of a code* is the minimum of $d(c_1, c_2)$ for all codewords c_1, c_2 with $c_1 \neq c_2$.

Lemma 5.3. *Let C be a code with minimum distance $d > 0$. Then*

(i) C is $(d-1)$-error detecting, but cannot detect all sets of d errors.

(ii) C is $\lfloor \frac{d-1}{2} \rfloor$-error correting, but cannot correct all sets of $\lfloor \frac{d-1}{2} \rfloor + 1$ errors.

Proof.

(i) If $x \in \mathbb{F}_2^n$ and $c \in C$ are such that $0 < d(x,c) \leq d-1$, then we know that $x \notin C$ so this is $(d-1)$-error detecting. However there must exist $c_1, c_2 \in C$ such that $d(c_1, c_2) = d$, so we cannot say if there's an error if c_1 is 'corrupted' to c_2 in d errors.

(ii) Take $e = \lfloor \frac{d-1}{2} \rfloor$. If $x \in \mathbb{F}_2^n$ and $c_1 \in C$ are such that $d(x, c_1) \leq \lfloor e$ then for any $c_1 \neq c_2 \in C$ we have $d(x, c_2) \geq d(c_1, c_2) - d(c_1, x) \geq d - e > e$. So C is e-error correcting. Now take $c_1, c_2 \in C$ with $d(c_1, c_2) = d$. Then take $x \in \mathbb{F}_2^n$ such that x differs from c_1 is prescisely $e+1$ places where c_1 and c_2 differ. Then $d(c_1, x) = e + 1$ and $d(x, c_2) = d - (e+1) \leq e + 1$. So C cannot be $(e+1)$-error correcting.

□

Definition. A $[n, m]$-code with minimum distance is called a $[n, m, d]$*-code.*

Notes.

- $m \leq 2^n$ with equality if and only if $C = \mathbb{F}_2^n$ (trivial code)
- $d \leq n$, with equality in case of the repitition code.

Example 5.2.

(i) Repitition code of length n is a $[n, 2, n]$-code, $(n-1)$-error detecting and $\lfloor \frac{n-1}{2} \rfloor$-error correcting.

(ii) Simple parity check code is a $[n, 2^{n-1}, 2]$-code, 1-error detecting and 0-error correcting.

(iii) Hamming's original code is 1-error correcting, implying $d \geq 3$. Also $0000000, 1110000$ are distance 3 apart, so $d = 3$. So this is a $[7, 16, 3]$-code and is 2-error detecting.

6 Covering estimates

Take $x \in \mathbb{F}_2^n$, $r \geq 0$. Then $\overline{B}(x,r) = \{y \in \mathbb{F}_2^n : d(x,y) \leq r\}$ is the *closed Hamming ball*. Denote $V(n,r) = |\overline{B}(x,r)| = \sum_{i=0}^{r} \binom{n}{i}$, the *volume*.

Lemma 6.1 (Hamming's bound)**.** *An e-error correcting code C of length n has*

$$|C| \leq \frac{2^n}{V(n,e)}.$$

Proof. C is e-error correcting so $\{B(c,e)\}_{c\in C}$ are pairwise disjoint balls, so $\sum_{c\in C} |B(c,e)| = |C|V(n,e) \leq |\mathbb{F}_2^n| = 2^n$. $\square$

Lemma 6.2. *A code C of length n that can correct e errors is perfect if $|C| = \frac{2^n}{V(n,e)}$. Equivalently, for all $x \in \mathbb{F}_2^n$ there exists a unique $c \in C$ such that $d(x,c) \leq e$. In this case, any $e+1$ errors will make you decode incorrectly.*

Example 6.1.

(a) Hamming $[7,16,3]$-code is 1-error correcting and

$$\frac{2^n}{V(n,e)} = \frac{2^7}{V(7,1)} = \frac{2^7}{1+7} = 2^4 = |C|.$$

(b) Binary repitition code of length n (for n odd) is perfect.

Remark. If $\frac{2^n}{V(n,e)} \notin \mathbb{Z}$ then there does not exist a perfect e-error correcting code of length n. Converse is also false ($n = 90, e = 2$ on Example Sheet 2).

Definition. Define $A(n,d) = \max\{m : \exists [n,m,d]\text{-code}\}$.

The $A(n,d)$ are unknown in general. But we have some special cases:

Examples.

- $A(n,1) = 2^n$ (trivial code)
- $A(n,n) = 2$ (repitition code)
- $A(n,2) = 2^{n-1}$ (simple parity check code)

Lemma 6.3. *$A(n,d+1) \leq A(n,d)$.*

Proof. Let $m = A(n,d+1)$ and take a $[n,m,d+1]$-code C. Let $c_1, c_2 \in C$ have $d(c_1,c_2) = d+1$. Let c_1' differ from c_1 in a single place where c_1 and c_2 differ. Hence $d(c_1', c_2) = d$. If $c \in C \setminus \{c_1\}$, then $d(c,c_1) \leq d(c,c_1') + d(c_1',c_1)$ so $d+1 \leq d(c,c_1') + 1$. Hence $d(c,c_1') \geq d$. So replacing c_1 with c_1', we get an $[n,m,d]$-code. $\square$

Corollary 6.4. *$A(n,d) = \max\{m : \exists [n,m,d']\text{-code for some } d' \geq d\}$.*

Theorem 6.5.

$$\frac{2^n}{V(n,d-1)} \underbrace{\leq}_{\textit{GSV bound}} A(n,d) \underbrace{\leq}_{\textit{Hamming bound}} \frac{2^n}{V\left(n, \lfloor \frac{d-1}{2} \rfloor\right)}.$$

Proof. We have already proved the Hamming bound. So let $m = A(n, d)$. Let C be a $[n, m, d]$-code. Then there does not exist $d(x, c) \geq d$ for all $c \in C$ (otherwise could replace C with $C \cup \{x\}$, contradicting maximality of m). Hence

$$\mathbb{F}_2^n \subseteq \bigcup_{c \in C} \overline{B}(c, d-1) \implies 2^n \leq \sum_{c \in C} |\overline{B}(c, d-1)| = mV(n, d-1).$$

□

Example 6.2. $n = 10$, $d = 3$, have $V(n, 1) = 11$, $V(n, 2) = 56$. The above theorem gives $19 \leq \frac{2^{10}}{56} \leq A(10, 3) \leq \frac{2^{10}}{11} \leq 93$. It was known that $72 \leq A(10, 3) \leq 93$, but the exact value of $A(10, 3)$ was only found in 1999.

Asymptotics of $V(n, r)$

We study $\frac{\log A(n, \lfloor n\delta \rfloor)}{n}$ as $n \to \infty$ to see how large the information rate can be for a given error rate.

Proposition 6.6. *Let $\delta \in (0, 1/2)$. Then*

(i) $\log V(n, \lfloor n\delta \rfloor) \leq nH(\delta)$*;*

(ii) $\frac{1}{n} \log A(n, \lfloor n\delta \rfloor) \geq 1 - H(\delta)$*.*

Where $H(\delta) = -\delta \log \delta - (1 - \delta) \log(1 - \delta)$*.*

Proof. First we show (i) ⇒ (ii): by the GSV bound,

$$A(n, \lfloor n\delta \rfloor) \geq \frac{2^n}{V(n, \lfloor n\delta \rfloor - 1)} \geq \frac{2^n}{V(n, \lfloor n\delta \rfloor)}$$

and so

$$\frac{\log A(n, \lfloor n\delta \rfloor)}{n} \geq 1 - \frac{\log V(n, \lfloor n\delta \rfloor)}{n} \geq 1 - H(\delta).$$

Now we prove (i): $H(\delta)$ is increasing for $\delta < 1/2$, so wlog we may assume $n\delta \in \mathbb{Z}$. Now

$$\begin{aligned} 1 = (\delta + (1 - \delta))^n = \sum_{i=0}^{n} \binom{n}{i} \delta^i (1 - \delta)^{n-i} &\geq \sum_{i=0}^{n\delta} \binom{n}{i} \delta^i (1 - \delta)^{n-i} \\ &= (1 - \delta)^n \sum_{i=0}^{n\delta} \binom{n}{i} \left(\frac{\delta}{1 - \delta}\right)^i \\ &\geq (1 - \delta)^n \sum_{i=0}^{n\delta} \binom{n}{i} \left(\frac{\delta}{1 - \delta}\right)^{n\delta} \\ &= \delta^{n\delta} (1 - \delta)^{n(1-\delta)} V(n, n\delta). \end{aligned}$$

Now taking logs:

$$0 \geq n(\delta \log \delta + (1 - \delta) \log(1 - \delta)) + \log V(n, n\delta).$$

□

The constant $H(\delta)$ in the above bound best possible:

Lemma 6.7. *We have*

$$\lim_{n\to\infty} \frac{\log V(n, \lfloor n\delta \rfloor)}{n} = H(\delta).$$

Proof. Exercise. $\square$

7 Constructing new codes from old

We're given C, a $[n, m, d]$-code. Can check the details in the following:

Examples.

1. The *parity check extension* C^+ is
$$\left\{\left(c_1, \dots, c_n, \sum_{i=1}^{n} c_i\right) : (c_1, \dots, c_n) \in C\right\}$$
Is a $[n+1, m, d']$-code with $d \leq d' \leq d+1$, depending on whether d is odd or even.

2. Fix $1 \leq i \leq n$. Deleting the ith digit from each codeword gives the *punctured code* C^-
$$\{(c_1, \dots, c_{i-1}, c_{i+1}, \dots, c_n) : (c_1, \dots, c_n) \in C\}.$$
If $d \geq 2$, then it is a $[n-1, m, d']$-code with $d-1 \leq d' \leq d$.

3. Fix $1 \leq i \leq n$, and $\alpha \in \mathbb{F}_2$. The *shortened code* C' is
$$\{(c_1, \dots, c_{i-1}, c_{i+1}, \dots, c_n) : (c_1, \dots, c_{i-1}, \alpha, c_{i+1}, \dots, c_n) \in C\}.$$
It has parameters $[n, m', d']$ with $d' \geq d$ and $m' \geq \frac{m}{2}$ for a suitable choice of α.

Shannon's Theorems

8 AEP and Shannon's first coding theorem

Definition. A *source* is a sequence of random variables $X_1, X_2, \dots$ taking values in some alphabet $\mathcal{A}$. A source is *Bernouilli* (*memoryless*) if $X_1, X_2, \dots$ are iid: write (X, X_n). A source $X_1, X_2, \dots$ is *reliably encodable at rate* r if there exists a sequence of subsets $(A_n)_{n\geq 1}$ with $A_n \subseteq \mathcal{A}^n$ such that:

1. $\lim_{n\to\infty} \frac{\log A_n}{n} = r$;
2. $\lim_{n\to\infty} \mathbb{P}((X_1, \dots, X_n) \in A_n) = 1$.

The *information rate* H of a source is the infimum of all reliable encoding rates. Exercise: $0 \leq H \leq \log|\mathcal{A}|$ with both bounds attainable.

Shannon's first coding theorem computes the information rate of certain sources, including Bernouilli sources.

Reminders from IA Probability:

We have a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. A discrete random variable X is a function $X : \Omega \to \mathcal{A}$. The probability mass function $p_x : \mathcal{A} \to [0,1]$ is defined by $x \mapsto \mathbb{P}(X = x)$. Can consider $p \circ X = p(X) : \Omega \to [0,1]$, a random varible taking values in $[0,1]$.

Given a source $X_1, X_2, \dots$ of random variables with values in $\mathcal{A}$, the probability mass function of $X^{(n)} = (X_1, \dots, X_n)$ is $p_{X^{(n)}}$ given by $(x_1, \dots, x_n) \mapsto \mathbb{P}((X_1, \dots, X_n) = (x_1, \dots, x_n))$. Since $p_{X^{(n)}} : \mathcal{A}^n \to [0,1]$ and $X^{(n)} : \Omega \to \mathcal{A}^n$, you can form $p(X^{(n)}) = p_{X^{(n)}} \circ X^{(n)} : \Omega \to [0,1]$.

Example 8.1. Let $\mathcal{A} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$. Suppose

$$X^{(2)} = \begin{cases} \mathrm{AB} & \text{with probability } 0.3 \\ \mathrm{AC} & \text{with probability } 0.1 \\ \mathrm{BC} & \text{with probability } 0.1 \\ \mathrm{BA} & \text{with probability } 0.2 \\ \mathrm{CA} & \text{with probability } 0.25 \\ \mathrm{CB} & \text{with probability } 0.05 \end{cases}.$$

Then

$$p(X^{(2)}) = \begin{cases} 0.3 & \text{with probability } 0.3 \\ 0.1 & \text{with probability } 0.2 \\ 0.2 & \text{with probability } 0.2 \\ 0.25 & \text{with probability } 0.25 \\ 0.05 & \text{with probability } 0.05 \end{cases}.$$

So some points are "lumped together".

Given a source $X_1, X_2, \dots$ *converges in probability* to a random variable L (possibly constant) if for all $\varepsilon > 0$, $\mathbb{P}(|X_n - L| > \varepsilon) \to 0$ as $n \to \infty$. We write $X_n \xrightarrow{\mathbb{P}} L$.

The *Weak Law of Large Numbers* (WLLN) says that if $(X; X_n)$ are iid real-valued random variables with finite expectation $\mathbb{E}X$, we have

$$\frac{1}{n} \sum_{i=1}^{n} X_i \xrightarrow{\mathbb{P}} \mathbb{E}X.$$

Example 8.2. If $X_1, X_2, \dots$ are iid Bernouilli, then $p(X_1), p(X_2), \dots$ are iid random variables and $p(X_1, \dots, X_n) = p(X_1) \dots p(X_n)$. Note

$$-\frac{1}{n} \log p(X_1, \dots, X_n) = -\frac{1}{n} \sum_{i=1}^{n} \log p(X_i) \xrightarrow{\mathbb{P}} -\mathbb{E}(-\log p(X_1)) = H(X_1) \text{ as } n \to \infty.$$

Lemma 8.1. *The information rate of a Bernouilli source $X_1, X_2, \dots$ is at most the expected word length of an optimal code $c : \mathcal{A} \to \{0,1\}^*$ for X_1.*

Proof. Let $l_1, l_2, \ldots$ be the lengths of codewords when we encode $X_1, X_2, \ldots$ using c. Let $\varepsilon > 0$. Set $A_n = \{x \in \mathcal{A}^n : c^*(x) \text{ has length at less than } n(\mathbb{E}l_1 + \varepsilon)\}$. Then

$$\begin{aligned}
\mathbb{P}((X_1, \ldots, X_n) \in A_n) &= \mathbb{P}\left(\sum_{i=1}^n l_i < n(\mathbb{E}l_1 + \varepsilon)\right) \\
&= \mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^n l_i - \mathbb{E}l_1\right| < \varepsilon\right) \\
&\to 1 \text{ as } n \to \infty.
\end{aligned}$$

Now, c is decipherable so c^* is injective. Hence $|A_n| \le 2^{n(\mathbb{E}l_1+\varepsilon)}$. Making A_n larger if necessary, $|A_n| = \lfloor e^{n(\mathbb{E}l_1+\varepsilon)} \rfloor$ so

$$\frac{\log(A_n)}{n} \to \mathbb{E}l_1 + \varepsilon.$$

Hence $X_1, X_2, \ldots$ is reliably encodable at rate $r = \mathbb{E}l_1 + \varepsilon$ for all $\varepsilon > 0$. Hence the information rate is at most $\mathbb{E}l_1$. □

Corollary 8.2. *A Bernouilli source has information rate less than $H(X_1) + 1$.*

Proof. Combine the above with the Noiseless Coding Theorem. □

We encode $X_1, X_2, \ldots$ in blocks

$$\underbrace{X_1, \ldots, X_N}_{Y_1}, \underbrace{X_{n+1}, \ldots, X_{2N}}_{Y_2}, \ldots$$

so $Y_1, Y_2, \ldots$ take values in $\mathcal{A}^N$. Exercise: show that if $X_1, X_2, \ldots$ has information rate H then $Y_1, Y_2, \ldots$ has information rate NH.

Proposition 8.3. *The information rate H of a Bernouilli source $X_1, X_2, \ldots$ is at most $H(X_1)$.*

Proof. Apply the previous corollary to $Y_1, Y_2, \ldots$ and obtain

$$NH < H(Y_1)+1 = H(X_1, \ldots, X_N)+1 = \sum_{i=1}^N H(X_i)+1 = NH(X_1, \ldots, X_n)+1.$$

Hence $H < H(X_1) + \frac{1}{N}$. Since N is arbitrary, $H \le H(X_1)$. □

Definition. A source $X_1, X_2, \ldots$ satisfies the *Asympotitic Equipartition Property* (AEP) for some constant $H \ge 0$ if

$$-\frac{1}{n}\log p(X_1, X_2, \ldots) \xrightarrow{\mathbb{P}} H \text{ as } n \to \infty.$$

Example 8.3. Tossing a biased coin, $\mathbb{P}(\mathrm{H}) = p$. Let $(X; X_n)$ be the results of independent coin tosses. After a large number N of tosses, expect on average pN heads and $(1-p)N$ tails. The probability of any particular sequence of pN heads and $(1-p)N$ tails is $p^{pN}(1-p)^{(1-p)N} = 2^{N(p \log p + (1-p)\log(1-p))} = 2^{-NH(X)}$. Not every sequence of tosses will be like this, but there is only a small probability of "atypical" sequences. With high probability we get a "typical" sequence and its probability will be close to $2^{-NH(X)}$.

Lemma 8.4. *The AEP for a source $X_1, X_2, \ldots$ is equivalent to the following property*

$$\forall \varepsilon > 0 \ \exists n_0(\varepsilon) \text{ such that } \forall n \geq n_0(\varepsilon) \ \exists \text{ a “typical set” } T_n \subseteq \mathcal{A}^n \text{ such that}$$

(i) $\mathbb{P}((X_1, \ldots, X_n) \in T_n) > 1 - \varepsilon$;

(ii) $2^{-n(H+\varepsilon)} \leq p(x_1, \ldots, x_n) \leq 2^{-n(H-\varepsilon)}$ *for all* $(x_1, \ldots, x_n) \in T_n$.

Proof. Obvious and non-examinable. □

Theorem 8.5 (Shannon's First Coding Theorem (FCT))**.** *If a source $X_1, X_2, \ldots$ satisfies the AEP with constant H, then the source has information rate H.*