

Coding and Cryptography

January 22, 2018

<i>CONTENTS</i>	2
-----------------	---

Contents

0	Miscellaneous	3
1	Introduction to communication channels and coding	4
1.1	Noiseless coding	5
1.2	Mathematical entropy	7

0 Miscellaneous

1 Introduction to communication channels and coding

For example, given a message $M = \text{"Callme!"}$ which we wish to send by email. We first encode it as binary strings using ASCII. So $f(C) = 1000011$, $f(a) = 1100001$, $f^*(M) = 10000111100001...0100001$.

The message goes from the source to the receiver after encoded by the source and decoded by the receiver via a channel, where errors could occur. The basic problem is, given a source and a channel (described probabilistically, we aim to design an encoder and a decoder in order to transmit information economically, reliably, and preserving privacy (secretly).

Some examples of each aspect:

economically: Morse code, where common letters have shorter codewords;

reliability: every book has an ISBN of form $a_1...a_{10}$ where $a_i \in \{0, 1, ..., 9\}$ for $1 \leq i \leq 9$ and $a_{10} \in \{0, 1, ..., 9, X\}$, s.t. $10a_1 + 9a_2 + ... + a_{10} \equiv 0 \pmod{11}$, where we treat X as 10. In this way errors can be detected, although not corrected. There is another version of ISBN which is 13 digit;

preserve privacy RSA.

A communication channel takes letters from an input alphabet $\Sigma_1 = \{a_1, ..., a_r\}$ and emits letters from an output alphabet $\Sigma_2 = \{b_1, ..., b_s\}$.

A channel is determined by the probabilities $P(y_1, ..., y_k \text{ received} | x_1, ..., x_k \text{ sent})$.

Definition. A *discrete memoryless channel* (DMC) is a channel for which $P_{ij} = P(b_j \text{ received} | a_i \text{ sent})$ is the same each time the channel is used, and is independent of all past and future. The channel matrix is the $r \times s$ matrix with entries p_{ij} . Note the rows sum to 1.

Example. (Binary Symmetric Channel, BSC)

BSC has $\Sigma_1 = \Sigma_2 = \{0, 1\}$, $0 \leq p \leq 1$. It has channel matrix $\begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$, i.e. p is the probability symbol is mistransmitted.

Example. (Binary Erasure Channel)

$\Sigma_1 = \{0, 1\}$, $\Sigma_2 = \{0, 1, *\}$, $0 \leq p \leq 1$. Then the channel matrix is $\begin{pmatrix} 1-p & p & 0 \\ 0 & p & 1-p \end{pmatrix}$, i.e. p is the probability that a symbol can't be read.

Informal definition: A channel's capacity is the highest rate at which information can be reliably transmitted over the channel. Here rate means the units of information per unit time (we want that high), and reliably means arbitrarily small error probability.

There are 3 sections:

- 1) Noiseless coding (data compression);
- 2) Error control codes;
- 3) Cryptography.

1.1 Noiseless coding

Notation. For Σ an alphabet that $\Sigma^* = \bigcup_{n \geq 0} \Sigma^n$ be the set of all finite strings of elements of Σ .

If $x = x_1 \dots x_r$, $y = y_1 \dots y_s$ are strings from Σ , write xy for the concatenation $x_1 \dots x_r y_1 \dots y_s$. Further, $|x_1 \dots x_r y_1 \dots y_s| = r + s$ the length of string.

Definition. Let Σ_1, Σ_2 be two alphabets. A *code* is a function $f : \Sigma_1 \rightarrow \Sigma_2^*$. The strings $f(x)$ for $x \in E$ are called *codewords*.

Example. (Greek five code)

$\Sigma_1 = \{\alpha, \beta, \dots, \omega\}$ (24 letters); $\Sigma_2 = \{1, 2, 3, 4, 5\}$ (more used). Now let $\alpha \rightarrow 11, \beta \rightarrow 12, \dots, \omega \rightarrow 54$.

Example. $\Sigma_1 = \{\text{all words in the dictionary}\}$, $\Sigma_2 = \{A, B, \dots, \text{space}\}$. Then $f = \text{'spell the word and a space.'}$

We sent a message $x_1, \dots, x_n \in \Sigma_1^*$ as $f(x_1)f(x_2)\dots f(x_n) \in \Sigma_2^*$, i.e. extend f to $f^* : \Sigma_1^* \rightarrow \Sigma_2^*$.

Definition. A code f is *decipherable* if f^* is injective, i.e. every string from Σ_2 arises from at most one message.

Note that f being injective is not enough. See this example:

Example. $\Sigma_1 = \{1, 2, 3, 4\}$, $\Sigma_2 = \{0, 1\}$, $f(1) = 0$, $f(2) = 1$, $f(3) = 00$, $f(4) = 01$. Then f is injective, but $f^*(312) = 0001 = f^*(114)$ so f^* is not decipherable.

Notation. If $|\Sigma_1| = m$, $|\Sigma_2| = a$, then we say f is an a -ary code of size m (in particular, if $a = 2$ we use the word binary).

Our aim is to construct decipherable codes with short word lengths.

Provided $f : \Sigma_1 \rightarrow \Sigma_2^*$ is injective, the following codes are always decipherable:

- (1) A *Block code* is a code with all codewords of the same length (eg Greek fire code);
- (2) In a *comma code* we reserve one letter from Σ_2 that is only used to signal the end of the codeword (example 2);
- (3) A *prefix-free* code is a code where no codeword is a prefix of another (If $x, y \in \Sigma_2^*$, x is a prefix of y if $y = xz$ for some $z \in \Sigma_2^*$).

Remark. (1) and (2) are special cases of (3).

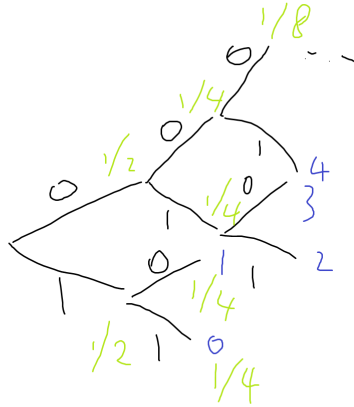
Prefix-free codes are also known as *instantaneous codes* (i.e. a word can be recognised as soon as its complete), or *self-punctuating codes*.

Theorem. (1.1, Kraft's inequality)

Let $|\Sigma_1| = m$, $|\Sigma_2| = a$. A prefix-free code $f : \Sigma_1 \rightarrow \Sigma_2^*$ with word lengths s_1, \dots, s_m exist iff

$$\sum_{i=1}^m a^{-s_i} \leq 1$$

Proof. First we prove forward implication. Consider an infinite tree where each has a descendent, labelled by the elements of Σ_2 . Each codeword corresponds to a node, the path from the root to this node spelling at the codeword. Assuming f is prefix-free, no codeword is the ancestor of any other. Now view the tree as a network with water being pumped in at constant rate and dividing the flow equally at each node. The total amount of water we can extract at the codewords is $\sum_{i=1}^m a^{-s_i}$ which is therefore ≤ 1 .



Conversely, suppose we can construct a prefix-free code with word lengths s_1, \dots, s_m , wlog $s_1 \leq s_2 \leq \dots \leq s_m$. We pick codewords of lengths s_1, \dots , sequentially ensuring previous codewords are not prefixes. Suppose there is no valid choice for the r^{th} codeword. The constructing the tree as above gives $\sum_{i=1}^{r-1} a^{-s_i} = 1$, contradicting our assumption. So we can construct a prefix-free code. \square

Theorem. (1.2, Mcmillan)

Every decipherable code satisfies Kraft's inequality.

Proof. (Karush)

Let $f : \Sigma_1 \rightarrow \Sigma_2^*$ be a decipherable code with word lengths s_1, \dots, s_m , let $s = \max_{1 \leq i \leq m} s_i$. Let $r \in \mathbb{N}$,

$$\left(\sum_{i=1}^m a^{-s_i} \right)^r = \sum_{b=1}^{rs} b_i a^{-l}$$

where b_i is the number of ways of choosing r codewords of total length l . f decipherable implies that $b_l \leq |\Sigma_2|^l = a^l$. Thus $\left(\sum_{i=1}^m a^{-s_i} \right)^r \leq \sum_{l=1}^{rs} a^l a^{-l} = rs$, so $\sum_{i=1}^m a^{-s_i} \leq (rs^{1/r}) \rightarrow 1$ as $r \rightarrow \infty$. So $\sum_{i=1}^m a^{-s_i} \leq 1$. \square

So we have a corollary: a decipherable code with prescribed word lengths exist iff there exists a prefix-free code with the same word lengths.

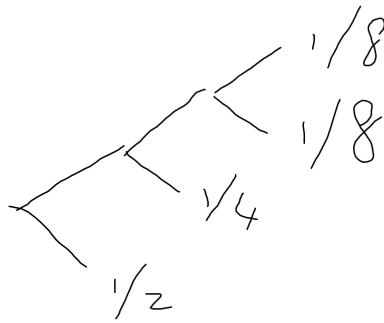
So we can restrict our attention to prefix-free codes.

1.2 Mathematical entropy

Entropy is a measure of 'randomness' or 'certainty'. Consider a random variable X taking values x_1, \dots, x_n with probability p_1, \dots, p_n ($\sum p_i = 1, 0 \leq p_i \leq 1$). The entropy $H(x)$ is roughly speaking the expected number of tosses of a fair coin needed to simulate X (or the expected number of yes/no questions we need to ask in order to establish the value of X).

Example. Suppose $p_1 = p_2 = p_3 = p_4 = \frac{1}{4}$. We identify $\{x_1, \dots, x_4\}$ with $\{HH, HT, TH, TT\}$, so $H(x) = 2$.

Example. $(p_1, p_2, p_3, p_4) = (1/2, 1/4, 1/8, 1/8)$. Then $H(x) = 1/2 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 = 7/4$. So the entropy here is smaller.



So in some sense, there is more randomness in the first example than the second.

Definition. (Entropy)

The entropy of X ,

$$H(X) = H(p_1, \dots, p_n) = - \sum_{i=1}^n p_i \log p_i$$

where, as most of the time in this course, $\log = \log_2$.

Remark. (1) If $p_i = 0$, we define $p_i \log p_i = 0$;
 (2) $H(X) \geq 0$.