NUS National University of Singapore

EE5139/EE6139
Homework 4

Semester I, AY2021/2022
Marco Tomamichel

Exercise 4.1 Infinite alphabet optimal code [all]

Let X be an i.i.d. random variable with an infinite alphabet, $\mathcal{X} = \{1, 2, 3, \ldots\}$. In addition, let $P(X = i) = 2^{-i}$.

a.) What is the entropy of X?

b.) Find an optimal variable length code, and show that it is indeed optimal.

Exercise 4.2 Codeword lengths for Huffmann codes [all]

(from 2019/2020 quiz)

Consider a random variable X which takes on four possible values with probabilities $(\frac{1}{3}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12})$.

a.) (5 points) Construct a Huffman code for this source.

b.) (5 points) Show that there exist two different sets of optimal lengths for the codewords, namely, show that codeword length assignments $(1, 2, 3, 3)$ and $(2, 2, 2, 2)$ are both optimal.

c.) (5 points) Are there optimal codes with codeword lengths for some symbols that exceed the Shannon code length $\lceil \log \frac{1}{p(x)} \rceil$?

Exercise 4.3 A better Morse code [EE5139]

In the previous exercise we designed a code for English letters that had slightly lower expected length than the Morse code. Now let us look at this problem a bit more closely. For the Morse code, the codeword symbol “-” requires 4 time units to send, whereas “.” only requires 2 time units. The end-of-letter symbol “_” requires 3 time units. So what we actually want to optimise is not the codeword length but the the *time* require to send it, e.g. for the codeword “.-._” this would be 11.

a.) Devise an algorithm that produces an alternative Morse code that optimises the expected time for a source producing English letters.

b.) Compute the expected time of transmission of the above code. Compute the expected time of the code produced in Exercise 3.3a, where we treat 0 as ‘.’ and 1 as ‘-’.

c.) Can you show that it is optimal?

d.) Can you come up with a prefix code that attempts to minimise the expected transmission time? This code does not need the end-of-letter symbol. Your algorithm can be heuristic (you do not need to prove optimality). Compute the expected transmission time and compare it to the code from a.

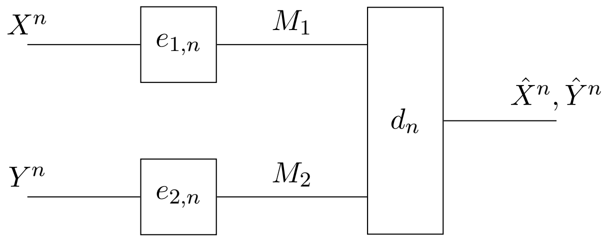
a	8.4%	b	1.5%	c	2.2%	d	4.2%	e	11.0%	f	2.2%	g	2.0%
h	6.0%	i	7.4%	j	0.1%	k	1.3%	l	4.0%	m	2.4%	n	6.7%
o	7.4%	p	1.9%	q	0.1%	r	7.5%	s	6.2%	t	9.2%	u	2.7%
v	0.9%	w	2.5%	x	0.1%	y	2.0%	z	0.1%				

Table 1: Statistical distribution of letters in the English language. Source: `https://en.wikipedia.org/wiki/Letter_frequency`, but normalized so that they add up to 100%.

Exercise 4.4 Converse for the Slepian–Wolf coding problem [all]

Let X and Y be a pair of jointly distributed random variables. (X is distributed on finite set $\mathcal{X}$, and Y is distributed on finite set $\mathcal{Y}$.) An $(n, 2^{nL_1}, 2^{nL_2})$-separately-encoded-jointly-decoded source code consists of a pair of encoders e_1, e_2, and a decoder d, where

- $e_1 : \mathcal{X}^n \to \{0,1\}^{nL_1}$,
- $e_2 : \mathcal{Y}^n \to \{0,1\}^{nL_2}$, and
- $d : \{0,1\}^{nL_1} \times \{0,1\}^{nL_2} \to \mathcal{X}^n \times \mathcal{Y}^n$.



The rate pair (R_1, R_2) is said to be achievable for DMS (X, Y) if there exists a sequence of $(n, 2^{nL_1}, 2^{nL_2})$-codes with encoders $e_{1,n}$, $e_{2,n}$ and decoder d_n such that

$$\lim_{n\to\infty} P\{(\hat{X}^n, \hat{Y}^n) \neq (X^n, Y^n)\} = 0$$

where

$$(\hat{X}^n, \hat{Y}^n) = d_n(M_1, M_2),\ M_1 = e_{1,n}(X^n),\ \text{and}\ M_2 = e_{2,n}(Y^n)$$

are the reconstructed source and codewords respectively.
Prove that, for any (R_1, R_2) achievable, it must hold that

$$R_1 \geq H(X|Y), \tag{1}$$
$$R_2 \geq H(Y|X), \tag{2}$$
$$R_1 + R_2 \geq H(X, Y). \tag{3}$$