EE5139/EE6139
Homework 3

Semester I, AY2021/2022
Marco Tomamichel

Exercise 3.1 Huffman code [EE5139]

Consider a source that outputs independent letters according to the frequency that they appear in the English language (use the frequencies listed in Table 1).

a.) Calculate the entropy of this source.

b.) Construct a Huffman code for this source. You may either construct the Huffman code manually according to the algorithm discussed in the lecture, or write a computer program that does this for you.

c.) Compute the expected length of the codeword for this code. How does this compare to the entropy computed in a)?

Exercise 3.2 Huffman code [all]

Which of the following sets of codewords can never be valid Huffman codes (for any assignment of probabilities)? Argue why.

a.) $\{0, 10, 11\}$,

b.) $\{00, 01, 10, 11\}$,

c.) $\{00, 01, 10, 110\}$,

d.) $\{01, 10\}$,

e.) $\{1, 01, 10\}$.

Exercise 3.3 A code that allows a prefix [EE5139]

As we have seen in the lecture, we like codes to be prefix-free as otherwise we do not know when a codeword ends and decoding might no longer be unique. One simple (but not necessarily very efficient) way to overcome this problem is to simply add a special symbol that indicates the end of a codeword. In this exercise we will thus consider codewords that are comprised of "0" and "1" and always end with a special space character "_".

a.) Construct a code for this source in the following way: the two most frequent laters are assigned codewords of length $1+1$, the next four most frequent letters codewords of lenght $2+1$, etc.

b.) Compute the expected length of the codewords.

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%.

c.) The code you arrived at is very similar to a very famous code that has been in use since the 1830s. Can you figure out which code that is? Compare the expected length of codewords of your code to that of this historical code.

Hint: Replace "0" by "·" and "1" by "–" in your codewords.

Exercise 3.4 Kraft–McMillan inequality for uniquely decodable codes [all]

Assume a uniquely decodable code has codeword lengths $l_1, \ldots, l_M$. Our goal is to derive Kraft's inequality for uniquely decodable codes:

$$\sum_{j=1}^{M} 2^{-l_j} \leq 1\,.$$

a.) Prove the following identity (this is easy):

$$\left(\sum_{j=1}^{M} 2^{-l_j}\right)^n = \sum_{j_1=1}^{M}\sum_{j_2=1}^{M}\cdots\sum_{j_n=1}^{M} 2^{-(l_{j_1}+l_{j_2}+\ldots+l_{j_n})}$$

b.) Let A_l be the number of concatenations of n codewords that have overall length $l = l_{j_1}+l_{j_2}+\ldots+l_{j_n}$ and let $l_{\max} = \max\{l_1, l_2, \ldots, l_M\}$ be the maximum length of a codeword. Show that

$$\left(\sum_{j=1}^{M} 2^{-l_j}\right)^n = \sum_{l=n}^{nl_{\max}} A_l\, 2^{-l}$$

c.) Using unique decodability, show that $A_i \leq 2^i$ and hence

$$\left(\sum_{j=1}^{M} 2^{-l_j}\right)^n \leq nl_{\max}$$

Use this to derive the desired inequality.

Exercise 3.5 Shannon code [all]

Let X be a random variable distributed on $\{0, 1, 2, \ldots, d-1\}$ for some $d > 1$, and let P_X be its probability mass function. Without loss of generality, we assume $P_X(0) \geqslant P_X(1) \geqslant \cdots \geqslant P_X(d-1) > 0$. A Shannon code for this source is constructed as follows. For each $x \in \mathcal{X}$, we consider the binary representation of the real number $\sum_{x'<x} P_X(x')$, and use the first $\lceil \log_2 \frac{1}{P_X(x)} \rceil$ bits in its fractional part to represent x.

Hint: For example, the binary representation of the real number $\frac{1}{3}$ is '0.01010101…', and the first 2 bits in its fractional part are '01'.

a.) Show that the above code is uniquely decodable.

b.) Produce a Shannon code table for the following source

x	0	1	2	3
$P_X(x)$	1/2	1/6	1/6	1/6

and compute the expected code length.

c.) Show that the expected length of the Huffman code of the above source is shorter.

Exercise 3.6 Huffman code algorithm [EE6139]

(from 2013/2014 final exam)

Consider a discrete memoryless source X with alphabet $\{1, 2, \ldots, M\}$. Suppose that the symbol probabilities are ordered and satisfy $p_1 > p_2 > \ldots > p_M$ and also satisfy $p_1 < p_{M-1} + p_M$. Let $l_1, l_2, \ldots, l_M$ be the lengths of a prefix-free code of minimum expected length for such a source.

(a) **(1 point)** Is the following statement true or false? $l_1 \leq l_2 \leq \ldots \leq l_M$. Argue why.

(b) **(2 points)** Show that if the Huffman algorithm is used to generate the above code, then $l_M \leq l_1 + 1$.

(c) **(2 points)** Show that $l_M \leq l_1 + 1$ for *any* (not necessarily Huffman generated) prefix-free code of minimum expected length. **Hint:** A minimum-expected-length code must be full.

(d) **(2 points)** Suppose $M = 2^k$ for some integer k. Show that all codewords must have the same length. **Hint:** What does the Kraft inequality look like for an optimal code? Consider the three cases $l_1 = k$, $l_1 < k$ and $l_1 > k$.