LINKÖPING UNIVERSITY

Solutions to Written Exam in
Data compression
TSBK08

8th June 2023

1 a) See the course literature.

b) See the course literature.

c) See the course literature.

d) See the course literature.

2 Start by showing that an optimal code for blocks of n symbols has a mean codeword length $\bar{l}$ that satisfies

$$H(X_1, X_2, \ldots, X_n) \leq \bar{l} < H(X_1, X_2, \ldots, X_n) + 1$$

See the course literature for how to prove this.

The mean data rate $R = \frac{\bar{l}}{n}$ will then satisfy

$$\frac{1}{n}H(X_1, X_2, \ldots, X_n) \leq R < \frac{1}{n}H(X_1, X_2, \ldots, X_n) + \frac{1}{n}$$

When n tends toward infinity, $\frac{1}{n}H(X_1, X_2, \ldots, X_n)$ will tend toward the entropy rate of the source, while $\frac{1}{n}$ will tend toward 0, showing that the data rate tends toward the entropy rate.

3 a) Stationary distribution $p(x_i, x_{i-1})$ for the states (pairs of symbols):

$$p(aa) = 0.5, \quad p(ab) = 0.0625, \quad p(ba) = 0.0625, \quad p(bb) = 0.375$$

From this distribution we get $H(X_i, X_{i-1}) \approx 1.5306$.

The marginal distribution gives us probabilities for single symbols

$$p(a) = p(aa) + p(ab) = 0.5625, \quad p(b) = p(ba) + p(bb) = 0.4375$$

From this distribution we get $H(X_i) \approx \underline{0.9887}$.

Finally, $H(X_i|X_{i-1}, X_{i-2})$ can be calculated as

$$\begin{aligned} H(X_i|X_{i-1}, X_{i-2}) &= 0.5 \cdot (-0.9 \cdot \log 0.9 - 0.1 \cdot \log 0.1) + \\ & \quad 0.0625 \cdot (-0.8 \cdot \log 0.8 - 0.2 \cdot \log 0.2) + \\ & \quad 0.0625 \cdot (-0.4 \cdot \log 0.4 - 0.6 \cdot \log 0.6) + \\ & \quad 0.375 \cdot (-0.1 \cdot \log 0.1 - 0.9 \cdot \log 0.9) \\ &\approx 0.5162 \end{aligned}$$

Alternatively, calculate the entropy $H(X_i, X_{i-1}, X_{i-2})$ and then $H(X_i|X_{i-1}, X_{i-2}) = H(X_i, X_{i-1}, X_{i-2}) - H(X_{i-1}, X_{i-2})$.

b) Probabilities for triples are given by $p(x_i, x_{i-1}, x_{i-2}) = p(x_{i-1}, x_{i-2}) \cdot p(x_i|x_{i-1}x_{i-2})$

$$p(aaa) = 0.45, \ p(baa) = 0.05, \ p(aab) = 0.05, \ p(bab) = 0.0125$$

$$p(aba) = 0.025, \ p(bba) = 0.0375, \ p(abb) = 0.0375, \ p(bbb) = 0.3375$$

A huffman code for triples can look like this

symbols	codeword	symbols	codeword
aaa	0	baa	1000
aab	1010	bab	10110
aba	10111	bba	10010
abb	10011	bbb	11

The average codeword length for this code is $\bar{l} = 2.0875$ bits/codeword, which gives the rate $R \approx 0.6958$ bits/symbol.

4 Assuming we use the alphabet order for the intervall order, with the a-interval closest to 0, the sequence corresponds to the intervall

$$[0.92401, \quad 0.928812)$$

with the interval size 0.004802. We will need at least $\lceil -\log_2 0.004802 \rceil = 8$ bits in our codeword, maybe one more.

Write the two interval limits as binary numbers:

$$\begin{aligned} 0.92401 &= 0.11101100100010\ldots \\ 0.928812 &= 0.11101101110001\ldots \end{aligned}$$

The smallest eight bit number inside the interval is 0.11101101, but there are numbers starting with these bits that are outside the interval (ie larger than the upper interval limit). Thus, we must use nine bits.

The codeword is **111011010**.

5 a) The search buffer size is $32 = 2^5$ and we thus need 5 bits to code the offset pointer. The alphabet size is 8, thus we will use $\log_2 8 = 3$ bits to code a symbol. If we code a single symbol we will use a total of $1+3=4$ bits and if we code a match we will use a total of $1+5+3=9$ bits. This means that it is better to code matches of length 1 and 2 as single symbols. Since we use 3 bits for the lengths, we can then code match lengths between 3 and 10 with our eight available codewords.

b)

f	o	l	c	codeword	sequence
0			t	0 110	t
0			u	0 111	u
0			r	0 100	r
1	2	5		1 00010 010	turtu
0			s	0 101	s
1	1	3		1 00001 000	usu
0			p	0 010	p
0			o	0 001	o
1	8	7		1 01000 100	rtususu
0			s	0 101	s
0			p	0 010	p
1	0	3		1 00000 000	ppp
1	12	3		1 01100 000	ort
0			n	0 000	n

6 Create all cyclic shifts of the sequence and sort them

Cyclic shifts	Sorted shifts
$susstuss$	$sssusstu$
$ssusstus$	$sstusssu$
$sssusstu$	$ssusstus$
$usssusst$	$stusssus$
$tusssuss$	$susstuss$
$stusssus$	$tusssuss$
$sstusssu$	$usssusst$
$usstusss$	$usstusss$

The result of the transform is the sequence $uussssts$ (the last column of the sorted list) and the position in the sorted list where we find the original sequence, ie 4 (assuming we start counting at 0).

The mtf coding gives the sequence 2,0,1,0,0,0,2,1.

7 The differential entropy is given by

$$\begin{aligned} h(X) &= -\int_{-\infty}^{\infty} f(x)\log f(x)dx = -2\left(\int_0^1 0.4\cdot\log 0.4\ dx + \int_1^2 0.1\cdot\log 0.1\ dx\right) \\ &= -0.8\cdot\log 4 - \log 0.1 = /\text{assuming base 2}/ = \log 10 - 1.6 \approx 1.7219 \text{ [bits]} \end{aligned}$$