

Department of Electrical and Computer Engineering
ECE 776 - Information Theory

Reading

Cover and Thomas, Chapter 5, 12.10.

Homework 4

1. Problem 5.4

(a) The binary Huffman code is

Symbol	Step 0	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Codeword
1	0.49	0.49	0.49	0.49	0.49	0.51	1	1
2	0.26	0.26	0.26	0.26	0.26	0.49		00
3	0.12	0.12	0.12	0.13	0.25			011
4	0.04	0.05	0.08	0.12				01000
5	0.04	0.04	0.05					01001
6	0.03	0.04						01010
7	0.02							01011

(b) The expected length is $E[L] = \sum_i p_i l_i = 2.02\text{bits}$.

(c) The ternary Huffman code is

Symbol	Step 0	Step 1	Step 2	Step 3	Codeword
1	0.49	0.49	0.49	1	0
2	0.26	0.26	0.26		1
3	0.12	0.12	0.25		20
4	0.04	0.09			22
5	0.04	0.04			210
6	0.03				211
7	0.02				212

2. **Problem 5.12**

(a) The binary Huffman code is

Symbol	Step 0	Step 1	Step 2	Step 3	Codeword
1	1/3	1/3	2/3	1	0
2	1/3	1/3	1/3		11
3	1/4	1/3			101
4	1/12				100

(b) Both set of codeword length $\{1, 2, 3, 3\}$ and $\{2, 2, 2, 2\}$ achieves the same expected length. Since Huffman is optimal, the alternative code is also optimal.

(c) The message x_4 with probability $1/4$ has Huffman code of length 3, longer than the Shannon codeword length $\lceil \log 1/p \rceil$. Thus, Huffman codeword for a particular symbol might be longer than Shannon codeword. However, after averaging, Huffman code is always optimal.

3. **Problem 5.22**

Notice that $p_1 > p_2$. Also, let $\{c_1, \dots, c_m\}$ denote the codewords with length $\{l_1, \dots, l_m\}$, corresponding to the probabilities $\{p_1, \dots, p_m\}$.

(a) Suppose $l_1 = 2$, corresponding to $c_1 = 00$. Then, since Huffman code is prefix free, there exist three sets of codewords starts with 01, 10, 11 correspondingly. Let C_{01}, C_{10}, C_{11} denote these sets of codewords with probabilities $P_a \geq P_b \geq P_c$. It is obvious that $p_1 = 2/5 > P_b \geq P_c$. Furthermore, it must be $P_b + P_c < 2/5$, because if we assume that $P_b + P_c \geq 2/5$, then $P_a \geq P_b \geq 1/5$ (why?) and $p_1 + P_a + P_b + P_c > 1$. Without loss of generality, let C_{10} and C_{11} denotes these two sets with probability P_b and P_c . Then, we can exchange the first bits of codewords in C_{10} and C_{11} to 00 and let $c'_1 = 1$. This new code has length

$$L' < 1 * 2/5 + \sum_{C_{01}} p_i l_i + \sum_{C_{10}} p_i (l_i + 1) + \sum_{C_{11}} p_i (l_i + 1) < 2 * 2/5 + \sum_{C_{01}} p_i l_i + \sum_{C_{10}} p_i l_i + \sum_{C_{11}} p_i l_i < L.$$

Thus, we have reached a contradiction.

(b) Suppose $l_1 = 1$, corresponding to $c_1 = 0$. Then, two sets of codewords starts with 10, 11 correspondingly. Let C_{10}, C_{11} denote these sets of codewords. Since the total probability of C_{10} and C_{11} is greater than $2/3 = 1 - 1/3$, at least one set must have probability greater than $1/3$. Without loss of generality, assum that set is C_{11} . We can change the first two bits of codewords in C_{11} to 0 and let c'_1 be 11. This code has length

$$L' < 2 * 1/3 + \sum_{C_{10}} p_i l_i + \sum_{C_{11}} p_i (l_i - 1) < 1 * 1/3 + \sum_{C_{10}} p_i l_i + \sum_{C_{11}} p_i l_i < L.$$

Thus, we have reached a contradiction.

4. **Problem 5.25**

(a) Since $l_i = \lceil \log \frac{1}{p_i} \rceil$, we have

$$\log \frac{1}{p_i} \leq l_i \leq \log \frac{1}{p_i} + 1,$$

which implies that

$$H(X) \leq E[L] = \sum p_i l_i \leq H(X) + 1.$$

To show the code is prefix free, notice that

$$2^{-l_i} \leq p_i \leq 2^{-(l_i-1)}.$$

For $j > i$, F_j differs from F_i by at least 2^{-l_i} . Therefore, F_j will differ from F_i at least one bit within the first l_i bits of binary expansion. So, F_i can not be prefix of F_j . Thus, the code is a prefix code.

(b) The Shannon-Fano-Elias code is

Symbol	Probability	F_i (decimal)	F_i (binary)	l_i	Codeword
1	0.5	0.0	0.0	1	0
2	0.25	0.5	0.10	2	10
3	0.125	0.75	0.110	3	110
4	0.125	0.875	0.111	3	111

5. Problem 12.12

We first parse the string, looking for new sub-strings. The parsing gives 0,00,000,1,10,101,0000,01,1010,1. These 10 phrases forms the prefix dictionary and needs 4-bit description. We encode the string as (0000,0), (0001,0), (0010,0), (0000,1), (0100,0), (0101,1),(0011,0), (0001,1), (0110,0), (0000,1). (The last phrase is handled like new phrase even if it is not.)