16:332:542
Information Theory and Coding
Problem Set 4 Solutions

Chapter 5: 8, 11, 12, 14, 22,25

Chapter 5:

8. *Simple optimum compression of a Markov source.* Consider the 3-state Markov process having transition matrix

$U_{n-1} \backslash U_n$	S_1	S_2	S_3
S_1	1/2	1/4	1/4
S_2	1/4	1/2	1/4
S_3	0	1/2	1/2

Thus the probability that S_1 follows S_3 is equal to zero. Design 3 codes C_1, C_2, C_3 (one for each state S_1, S_2, S_3), each code mapping elements of the set of S_i's into sequences of 0's and 1's, such that this Markov process can be sent with maximal compression by the following scheme:

(a) Note the present symbol S_i.

(b) Select code C_i.

(c) Note the next symbol S_j and send the codeword in C_i corresponding to S_j.

(d) Repeat for the next symbol.

What is the average message length of the next symbol conditioned on the previous state $S = S_i$ using this coding scheme? What is the unconditional average number of bits per source symbol? Relate this to the entropy rate $\mathcal{H}$ of the Markov chain.

Solution: *Simple optimum compression of a Markov source.*

It is easy to design an optimal code for each state. A possible solution is

Next state	S_1	S_2	S_3	
Code C_1	0	10	11	$E(L\|C_1) = 1.5$ bits/symbol
code C_2	10	0	11	$E(L\|C_2) = 1.5$ bits/symbol
code C_3	-	0	1	$E(L\|C_3) = 1$ bit/symbol

The average message lengths of the next symbol conditioned on the previous state being S_i are just the expected lengths of the codes C_i. Note that this code assignment achieves the conditional entropy lower bound.

---

11. *Suffix condition.* Consider codes that satisfy the suffix condition, which says that no codeword is a suffix of any other codeword. Show that a suffix condition code is uniquely decodable, and show that the minimum average length over all codes satisfying the suffix condition is the same as the average length of the Huffman code for that random variable.

Solution: *Suffix condition.* The fact that the codes are uniquely decodable can be seen easily be reversing the order of the code. For any received sequence, we work backwards from the end, and look for the reversed codewords. Since the codewords satisfy the suffix condition, the reversed codewords satisfy the prefix condition, and the we can uniquely decode the reversed code.

The fact that we achieve the same minimum expected length then follows directly from the results of Section 5.5. But we can use the same reversal argument to argue that corresponding to every suffix code, there is a prefix code of the same length and vice versa, and therefore we cannot achieve any lower codeword lengths with a suffix code than we can with a prefix code.

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

(a) Construct a Huffman code for this random variable.

(b) 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) Conclude that there are optimal codes with codeword lengths for some symbols that exceed the Shannon code length $\lceil \log \frac{1}{p(x)} \rceil$.

Solution: *Shannon codes and Huffman codes.*

(a) Applying the Huffman algorithm gives us the following table

Code	Symbol	Probability			
0	1	1/3	1/3	2/3	1
11	2	1/3	1/3	1/3	
101	3	1/4	1/3		
100	4	1/12			

which gives codeword lengths of 1,2,3,3 for the different codewords.

(b) Both set of lengths 1,2,3,3 and 2,2,2,2 satisfy the Kraft inequality, and they both achieve the same expected length (2 bits) for the above distribution. Therefore they are both optimal.

(c) The symbol with probability $1/4$ has an Huffman code of length 3, which is greater than $\lceil \log \frac{1}{p} \rceil$. Thus the Huffman code for a particular symbol may be longer than the Shannon code for that symbol. But on the average, the Huffman code cannot be longer than the Shannon code.

---

14. *Huffman code.* Find the (a) *binary* and (b) *ternary* Huffman codes for the random variable X with probabilities

$$p = \left(\frac{1}{21}, \frac{2}{21}, \frac{3}{21}, \frac{4}{21}, \frac{5}{21}, \frac{6}{21}\right) .$$

(c) Calculate $L = \sum p_i l_i$ in each case.

Solution: *Huffman code.*

(a) The Huffman tree for this distribution is

Codeword							
00	x_1	6/21	6/21	6/21	9/21	12/21	1
10	x_2	5/21	5/21	6/21	6/21	9/21	
11	x_3	4/21	4/21	5/21	6/21		
010	x_4	3/21	3/21	4/21			
0110	x_5	2/21	3/21				
0111	x_6	1/21					

(b) The ternary Huffman tree is

Codeword					
1	x_1	6/21	6/21	10/21	1
2	x_2	5/21	5/21	6/21	
00	x_3	4/21	4/21	5/21	
01	x_4	3/21	3/21		
020	x_5	2/21	3/21		
021	x_6	1/21			
022	x_7	0/21			

(c) The expected length of the codewords for the binary Huffman code is $51/21 = 2.43$ bits.

The ternary code has an expected length of $34/21 = 1.62$ ternary symbols.

---

22. *Optimal codeword lengths.* Although the codeword lengths of an optimal variable length code are complicated functions of the message probabilities $\{p_1, p_2, \ldots, p_m\}$, it can be said that less probable symbols are encoded into longer codewords. Suppose that the message probabilities are given in decreasing order $p_1 \geq p_2 \geq \cdots \geq p_m$.

(a) Prove that for any binary Huffman code, if the most probable message symbol has probability $p_1 > 2/5$, then that symbol must be assigned a codeword of length 1.

(b) Prove that for any binary Huffman code, if the most probable message symbol has probability $p_1 < 1/3$, then that symbol must be assigned a codeword of length ≥ 2.

Solution: *Optimal codeword lengths.* Let $\{c_1, c_2, \ldots, c_m\}$ be codewords of respective lengths $\{\ell_1, \ell_2, \ldots, \ell_m\}$ corresponding to probabilities $\{p_1, p_2, \ldots, p_m\}$.

(a) We prove that if $p_1 > p_2$ and $p_1 > 2/5$ then $\ell_1 = 1$. Suppose, for the sake of contradiction, that $\ell_1 \geq 2$. Then there are no codewords of length 1; otherwise c_1 would not be the shortest codeword. Without loss of generality, we can assume that c_1 begins with 00. For $x, y \in \{0,1\}$ let C_{xy} denote the set of codewords beginning with xy. Then the sets C_{01}, C_{10}, and C_{11} have total probability $1 - p_1 < 3/5$, so some two of these sets (without loss of generality, C_{10} and C_{11}) have total probability less 2/5. We can now obtain a better code by interchanging the subtree of the decoding tree beginning with 1 with the subtree beginning with 00; that is, we replace codewords of the form $1x\ldots$ by $00x\ldots$ and codewords of the form $00y\ldots$ by $1y\ldots$. This improvement contradicts the assumption that $\ell_1 \geq 2$, and so $\ell_1 = 1$. (Note that $p_1 > p_2$ was a hidden assumption for this problem; otherwise, for example, the probabilities $\{.49, .49, .02\}$ have the optimal code $\{00, 1, 01\}$.)

(b) The argument is similar to that of part (a). Suppose, for the sake of contradiction, that $\ell_1 = 1$. Without loss of generality, assume that $c_1 = 0$. The total probability of C_{10} and C_{11} is $1 - p_1 > 2/3$, so at least one of these two sets (without loss of generality, C_{10}) has probability greater than 2/3. We can now obtain a better code by interchanging the subtree of the decoding tree beginning with 0 with the subtree beginning with 10; that is, we replace codewords of the form $10x\ldots$ by $0x\ldots$ and we let $c_1 = 10$. This improvement contradicts the assumption that $\ell_1 = 1$, and so $\ell_1 \geq 2$.

---

25. *Shannon code.* Consider the following method for generating a code for a random variable X which takes on m values $\{1, 2, \ldots, m\}$ with probabilities $p_1, p_2, \ldots, p_m$.

Assume that the probabilities are ordered so that $p_1 \geq p_2 \geq \cdots \geq p_m$. Define

$$F_i = \sum_{k=1}^{i-1} p_i, \tag{5.37}$$

the sum of the probabilities of all symbols less than i. Then the codeword for i is the number $F_i \in [0,1]$ rounded off to l_i bits, where $l_i = \lceil \log \frac{1}{p_i} \rceil$.

(a) Show that the code constructed by this process is prefix-free and the average length satisfies

$$H(X) \leq L < H(X) + 1. \tag{5.38}$$

(b) Construct the code for the probability distribution $(0.5, 0.25, 0.125, 0.125)$.

Solution: *Shannon code.*

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

$$\log \frac{1}{p_i} \leq l_i < \log \frac{1}{p_i} + 1 \tag{5.39}$$

which implies that

$$H(X) \leq L = \sum p_i l_i < H(X) + 1. \tag{5.40}$$

The difficult part is to prove that the code is a prefix code. By the choice of l_i, we have

$$2^{-l_i} \leq p_i < 2^{-(l_i-1)}. \tag{5.41}$$

Thus F_j, $j > i$ differs from F_i by at least 2^{-l_i}, and will therefore differ from F_i is at least one place in the first l_i bits of the binary expansion of F_i. Thus the codeword for F_j, $j > i$, which has length $l_j \geq l_i$, differs from the codeword for F_i at least once in the first l_i places. Thus no codeword is a prefix of any other codeword.

(b) We build the following table

Symbol	Probability	F_i in decimal	F_i in 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

The Shannon code in this case achieves the entropy bound (1.75 bits) and is optimal.