Asymptotic Equipartition Property and Data Compression Exercises

Exercise 3.3:
The AEP and source coding. A discrete memoryless source emits a sequence of statistically independent binary digits with probabilities $p(1) = 0.005$ and $p(0) = 0.995$. The digits are taken 100 at a time and a binary codeword is provided for every sequence of 100 digits containing three or fewer ones.

(a) Assuming that all codewords are the same length, find the minimum length required to provide codewords for all sequences with three or fewer ones.

(b) Calculate the probability of observing a source sequence for which no codeword has been assigned.

Solution:

(a) The number of sequences of 100 digits containing three or few ones is given by

$$\begin{aligned} N &= \binom{100}{0} + \binom{100}{1} + \binom{100}{2} + \binom{100}{3} \\ &= 1 + 100 + 4980 + 161700 \\ &= 166751 \end{aligned} \tag{1}$$

The minimum length required to encode these sequences is given by $\lceil \log_2 N \rceil = \lceil 17.34731 \rceil = 18$.

(b) The probablity of observing a sequence which has an assigned codeword is given by:

$$\begin{aligned} P &= 1 \cdot 0.995^{100} + 100 \cdot 0.995^{99} \cdot 0.005 + 4980 \cdot 0.995^{98} \cdot 0.005^{2} + 161700 \cdot 0.995^{97} \cdot 0.005^{3} \\ &= 0.9983 \end{aligned} \tag{2}$$

Hence the probability of observing a sequence which has no codeword is 0.0017.

Exercise 5.4:
Huffman Coding. Consider the random variable

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 0.49 & 0.26 & 0.12 & 0.04 & 0.04 & 0.03 & 0.02 \end{pmatrix} \tag{3}$$

(a) Find a binary Huffman code for $\mathbf{X}$.

(b) Find the expected codelength for this encoding.

(c) Find a ternary Huffman code for $\mathbf{X}$ (a ternary code is one which uses three symbols, e.g. $\{0, 1, 2\}$, instead of a binary code's two symbols $\{0, 1\}$).

Solution:

(a) Using the diagram in Figure 1, the Huffman code for $\mathbf{X}$ is given in Table 1.

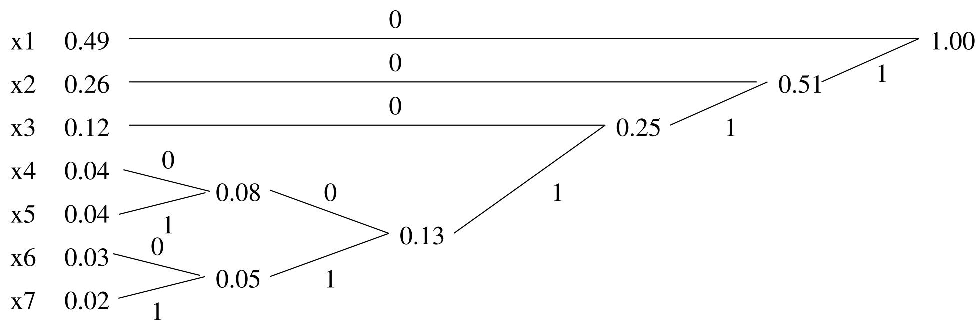

Figure 1: Diagram for designing the binary Huffman code for $\mathbf{X}$ in Exercise 5.4.

Table 1: Binary Huffman code for $\mathbf{X}$ in Exercise 5.4

X	**Code**
x_1	0
x_2	10
x_3	110
x_4	11100
x_5	11101
x_6	11110
x_7	11111

(b) The expected codelength for this encoding is:

$$\begin{aligned} E[L_x] &= 0.49 \times 1 + 0.26 \times 2 + 0.12 \times 3 + (0.04 + 0.04 + 0.03 + 0.02) \times 5 \\ &= 2.02 \end{aligned} \tag{4}$$

(c) Using the diagram in Figure 2, the ternary Huffman code for $\mathbf{X}$ is given in Table 2.

Exercise from Lectures:
Fano and Huffman codes. Construct Fano and Huffman codes for $\{0.2, 0.2, 0.18, 0.16, 0.14, 0.12\}$. Compare the expected number of bits per symbol in the two codes with each other and with the entropy. Which code is best?

Solution:
Using the diagram in Figure 3, the Fano code is given in Table 3. The expected codelength for the Fano code is:

$$\begin{aligned} E[L] &= (0.2 + 0.16) \times 2 + (0.2 + 0.18 + 0.14 + 0.12)) \times 3 \\ &= 2.64 \end{aligned} \tag{5}$$

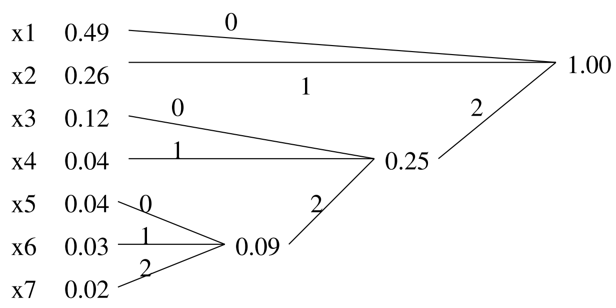

Figure 2: Diagram for designing the ternary Huffman code for **X** in Exercise 5.4.

Table 2: Ternary Huffman code for **X**

X	**Code**
x_1	0
x_2	1
x_3	20
x_4	21
x_5	220
x_6	221
x_7	222

Using the diagram in Figure 4, the Huffman code is given in Table 4. The expected codelength for the Huffman code is:

$$\begin{aligned} E[L] &= (0.2 + 0.2) \times 2 + (0.18 + 0.16 + 0.14 + 0.12) \times 3 \\ &= 2.6 \end{aligned} \tag{6}$$

The entropy is calculate as:

$$\begin{aligned} H &= -(0.2 \log 0.2 + 0.2 \log 0.2 + 0.18 \log 0.18 + 0.16 \log 0.16 + 0.14 \log 0.14 + 0.12 \log 0.12) \\ &= 2.56 \end{aligned} \tag{7}$$

Comparing the expected codelengths with the entropy, the Huffman code is the best code and achieves and expected codelength that is closest to the entropy.

Exercise 5.21:
Optimal codes for uniform distributions. Consider a random variable with m equiprobable outcomes. The entropy of this information sources is obviously $\log_2 m$ bits.

(a) Describe the optimal instantaneous binary code for this source and compute the average codeword length L_m.

(b) For what values of m does the average codeword length L_m equal the entropy $H = \log_2 m$?

(c) We know that $L < H + 1$ for any probability distribution. The redundancy of a variable length code is defined to be $\rho = L - H$. For what value(s) of m, where $2^k \leq m \leq 2^{k+1}$, is the redundancy of the code maximised? What is the limiting value of this worst case redundancy as $m \to \infty$?

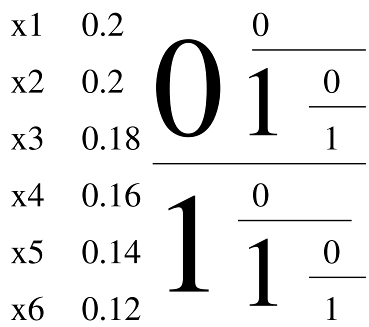

Figure 3: Diagram for designing the Fano code in the exercise from the lectures.

Table 3: Fano code for exercise from the lectures

X	**Code**
x_1	00
x_2	010
x_3	011
x_4	10
x_5	110
x_6	111

Solution:

(a) The optimal instantaneous binary code has codewords that differ by at most one bit. If d is difference between the number of outcomes m and the smallest power of 2,

$$d = m - 2^{\lfloor \log m \rfloor} \tag{8}$$

then there will be $2d$ codewords of length $\lceil \log m \rceil$ and $m - 2d$ codewords of length $\lfloor \log m \rfloor$. Let $b = \lfloor \log_2 m \rfloor$. When $m = 2^b$, every code is b bits long. For each new code required (i.e. for each increment in m) one b bit code has to be extended by one bit to make two $b+1$ bit codes, one for the old symbol coded by that b bit code and one for newly introduced symbol. Thus every increment in m leads to the removal of one b bit code and the introduction of two $b+1$ bit codes. If $d = m - 2^b$ then there will thus be $2d$ code words of length $b+1$ and $m - 2d$ code words of length b.

The average codeword length is given by:

$$\begin{aligned} L_m &= \frac{1}{m}\left(2d\lceil \log m \rceil + (m - 2d)\lfloor \log m \rfloor\right) \\ &= \frac{1}{m}\left(m\lfloor \log m \rfloor + 2d\right) \\ &= \lfloor \log m \rfloor + \frac{2d}{m} \end{aligned} \tag{9}$$

(b) The average codeword equals the entropy when m is a power of 2.

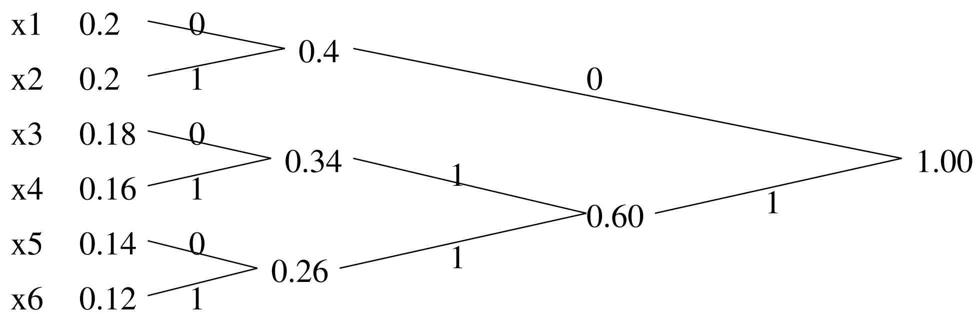

Figure 4: Diagram for designing the Huffman code in the exercise from the lectures.

Table 4: Binary Huffman code for **X**

X	**Code**
x_1	00
x_2	01
x_3	100
x_4	101
x_5	110
x_6	111

(c) When $m = 2^n + d$, the redundancy $\rho = L - H$ is given by

$$\begin{aligned}\rho &= L - \log m \\ &= \lfloor \log m \rfloor + \frac{2d}{m} - \log m \\ &= n + \frac{2d}{2^n + d} - \log(2^n + d) \\ &= n + \frac{2d}{2^n + d} - \frac{\ln(2^n + d)}{\ln 2}\end{aligned} \tag{10}$$

Differentiating with respect to d, we have

$$\frac{\partial \rho}{\partial d} = \frac{(2^n + 2d) \cdot 2 - 2d}{(2^n + d)^2} - \frac{1}{\ln 2} \cdot \frac{1}{2^n + d} \tag{11}$$

and setting this to zero, means that $d^* = 2^n(2\ln 2 - 1)$. Substituting this back into the equation for the redundancy, means that we have

$$\begin{aligned}\rho^* &= n + \frac{2d}{2^n + d} - \frac{\ln(2^n + d)}{\ln 2} \\ &= n + \frac{2 \cdot 2^n(2\ln 2 - 1)}{2^n + 2^n(2\ln 2 - 1)} - \frac{\ln(2^n + 2^n(2\ln 2 - 1))}{\ln 2} \\ &= 0.0861\end{aligned} \tag{12}$$

Exercise 5.25:
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

$$F_i = \sum_{k=1}^{i-1} p_k, \tag{13}$$

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

$$H(X) \leq L < H(X) + 1 \tag{14}$$

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

Solution:

(a) We look at the size of the increments to F_i. Since $l_i = \lceil \log \frac{1}{p_i} \rceil$, this means that

$$\begin{aligned} l_i - 1 &< \log \frac{1}{p_i} \leq l_i \\ 2^{l_i - 1} &< \frac{1}{p_i} \leq 2^{l_i} \\ 2^{-l_i} &\leq p_i < 2^{-l_i + 1} \end{aligned} \tag{15}$$

Since $l_i = \lceil \log \frac{1}{p_i} \rceil$,

$$\begin{aligned} &\log \frac{1}{p_i} \leq l_i < \log \frac{1}{p_i} + 1 \\ &p_i \log \frac{1}{p_i} \leq p_i l_i < p_i \log \frac{1}{p_i} + p_i \\ &\sum_i p_i \log \frac{1}{p_i} \leq \sum_i p_i l_i < \sum_i p_i \log \frac{1}{p_i} + \sum_i p_i \\ &H(X) \leq L(X) < H(X) + 1 \end{aligned} \tag{16}$$

Let x_k be the code word for symbol k.

x_k cannot be a prefix for x_i, $i < k$ because $l_i \leq l_k$ (N.B. if $l_i = l_k$ then there is the possibility that x_i and x_k could be identical, but this is covered by the following case by swapping the roles of i and k).

Let us now do a proof by contradication that x_k cannot be a prefix for x_{k+j}.

Assume x_k is a prefix of x_{k+j}.

Then x_k and x_{k+j} must agree in their first l_k bits.

Therefore $F_{k+j} - F_k < 2^{-l_k}$.

$$
\begin{aligned}
F_{k+j} - F_k &< 2^{-l_k} \\
\Rightarrow \sum_{i=1}^{k+j-1} p_i - \sum_{i=1}^{k-1} p_i &< 2^{-l_k} \\
\Rightarrow \sum_{i=k}^{k+j-1} p_i &< 2^{-l_k} \\
\Rightarrow p_k &< 2^{-l_k}
\end{aligned}
$$

But we know:

$$
\begin{aligned}
l_k &= \left\lceil \log_2 \frac{1}{p_k} \right\rceil \\
\Rightarrow l_k &\geq \log_2 \frac{1}{p_k} \\
\Rightarrow 2^{l_k} &\geq \frac{1}{p_k} \\
\Rightarrow 2^{-l_k} &\leq p_k
\end{aligned}
$$

This is a contradiction, therefore x_k cannot be a prefix for x_{k+j}, therefore the Shannon code is a prefix code.

(b) The code is designed as in Table 5:

Table 5: Shannon code for $\mathbf{X}$

i	p_i	$\lceil \log \frac{1}{p_i} \rceil$	F_i	Codeword
1	0.5	1	$0_{10} = 0.0_2$	0
2	0.25	2	$0.5_{10} = 0.1_2$	10
3	0.125	3	$0.75_{10} = 0.11_2$	110
4	0.125	3	$0.875_{10} = 0.111_2$	111