Example: *A code table for certain binary code is given as*

X_i	$P(X_i)$	Codeword	L_i
X_1	0.2	0	1
X_2	0.1	10	2
X_3	0.4	110	3
X_4	0.3	111	3

a) Find the average code length ; b) If a received sequence is 1011000111110, check if this sequence can be uniquely decoded or not.

Solution:-

a) $Lc = \bar{l}i = \sum_{i=1}^{n} li\, P(xi)$ = [1×0.2+2×0.1+3×0.4+3×0.3 = 2.5 bit / symbol

b) Using previous code table

10	110	0	0	111	110	
X_2	X_3	X_1	X_1	X_4	X_3 …	Decoded uniquely as $X_2X_3X_1X_1X_4X_3$.

Note that for previous example, the code is not optimum in terms of ***Lc***. we can reduce ***Lc*** by giving less L_i for X_i with higher probability. Hence the previous example can be modified.

This will give ***Lc=1.9*** bits/message which is less than before.

X_i	$P(X_i)$	Codeword	L_i
X_3	0.4	0	1
X_4	0.3	10	2
X_1	0.2	110	3
X_2	0.1	111	3

The condition for uniquely decodable code is that if x_i is given a codeword C_i of L_i bits then these L_i bits must not be the beginning of any other codeword C_i of higher length L_i for message x_i. So for previous example if "0" is a codeword, then no other codewords of higher length starts at "0". Also if "10" is a codeword, then no other codewords of higher length starts at "10" and so on. Hence the previous example was uniquely decodable, but the following codes

$x1$	0
$x2$	01
$x3$	10
$x4$	111

are not uniquely decodable since "0" is a codeword for x_1 while the codeword for x_2 starts with "0".

Code Efficiency & Redundancy:

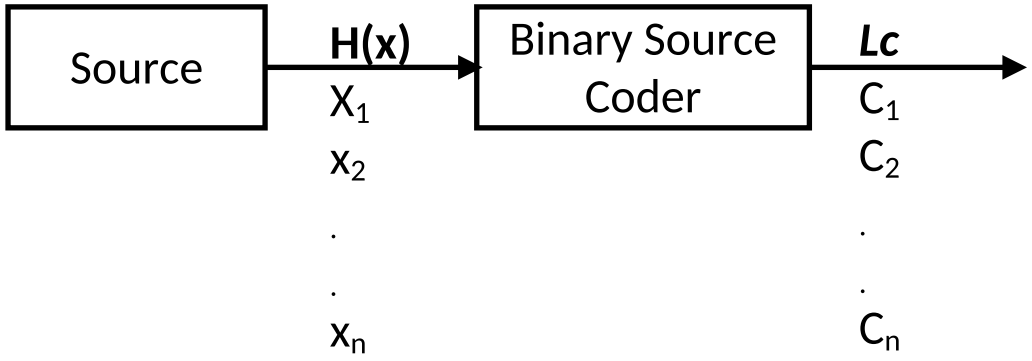
A code with average code length ***Lc*** has coding efficiency

$$\eta = \frac{H(x)}{Lc\ Log2D}$$

$\mathbf{D}$=2 for binary and $\mathbf{D}$=3 for Ternary and so on…

Redundancy R=1- η

For binary coding $\eta = \frac{H(x)}{Lc}$



Fixed Length Code: This is used when the source produces almost equiprobable messages $P(x1) \cong P(x2) \cong P(x3) .... \cong P(x_n)$ then $L_1 = L_2 .... L_n = Lc$.

For binary coding:

1. $Lc = \log_2(n)$ bit/symbol if $n = 2^r$, $r = 1,2,3,4$ and $n = 2,4,8,16$ which gives $\eta = 100\ \%$
2. $Lc = \text{int}[\log_2 n]+1$ if $n \neq 2^r$ which gives less efficiency

Example: For ten equiprobable messages coded in fixed code length. Find: ***Lc***, codeword and ***η?***

Solution:-

$n = 10$

$Lc = \text{int}[\log_2 10]+1 = 4$ bit/message

$H(X) = \log_2 10 = 3.3219$ bit/message

$\eta = H(x)/Lc = 3.3219/4 = 83\%$.

Xi	Codeword
X_1	0000
X_2	0001
X_3	0010
X_4	0011
X_5	0100
X_6	0101
X_7	0110
X_8	0111
X_9	1000
X_{10}	1001

Example: Find coding efficiency of a fixed code length used to encode messages obtained from throwing a fair dice: **a.** Once **b.** Twice **c.** 3-times

Solution:- For a fair dice, ***n***=6 equiprobable

a. Once

$Lc = \text{int}[\log_2 6]+1 = 3$ bit/message

$H(X) = \log_2 6 = 2.584$ bit/message

$\eta = H(x)/Lc = 2.584/3 = 86.1\%$

b. Twice

$n = 6 \times 6 = 36$

$Lc = \text{int}[\log_2 36]+1 = 6$ bit/message

$H(X) = \log_2 36 = 5.1699$

$\eta = H(x)/Lc = 5.1699/6 = 86.1\%$

c. Three times

$n = 6 \times 6 \times 6 = 216$

$Lc = \text{int}[\log_2 216]+1 = 8$ bit/message

$H(X) = \log_2 216 = 7.75488$

$\eta = H(x)/Lc = 7.75488/8 = 96.936\%$

When the messages probability are not equal then we use variable length codes, these codes are some times called (**minimum redundancy codes**). In the following, the Shannon, Fano and Huffman coding will be explained for binary coding (***D***=2) and ternary coding (***D***=3) will be easily modified.

1. Shannon Codes

For messages $X_1, X_2, \ldots\ldots, X_n$ with prob. $P(X_1), P(X_2) \ldots., P(X_n)$ then:-

$$l_i = \begin{cases} -log2\, p(xi) \ldots. & if p(xi) = \left(\frac{1}{2}\right)^r = (\frac{1}{2}, \frac{1}{4}, \frac{1}{8} \ldots) \\ int\,[-log2\, p(xi)] + 1 & \ldots. if p(xi) \neq \left(\frac{1}{2}\right)^r \end{cases}$$

Also define $\omega i = \sum_{k}^{i-1} p(xi)$ $\quad 0 \leq \omega i < 1$

Then the codeword of $\boldsymbol{x_i}$ is the binary equivalent for (D=2) for ωi consisting of l_i bits.

Before starting encoding, messages must be re-arranged in a decreasing order of probabilities.

Example: develop the Shannon code for the following set of messages, then find:
a) Code efficiency and b) P(0) & P(1) at the encoder output?
$P(\boldsymbol{x_i})=[0.4 \quad 0.25 \quad 0.15 \quad 0.1 \quad 0.07 \quad 0.03]$

Solution:

L_1 = int [$-\log_2 0.4$]+1 = 2 bit/message
$L_2 = -\log_2 0.25$= 2 bit/message
L_3 = int [$-\log_2 0.15$]+1 = 3 bit/message
L_4 = int [$-\log_2 0.1$]+1 = 4 bit/message
L_5 = int [$-\log_2 0.07$]+1 = 4 bit/message
L_6 = int [$-\log_2 0.03$]+1 = 6 bit/message

x_i	$P(x_i)$	ω_i	L_i	Codeword c_i	0_i	1_i
x_1	0.4	0	2	00	2	0
x_2	0.25	0.4	2	01	1	1
x_3	0.15	0.65	3	101	1	2
x_4	0.1	0.8	4	1100	2	2
x_5	0.07	0.9	4	1110	1	3
x_6	0.03	0.97	6	111110	1	5

$\omega_1 = 0$; $\quad \omega_2 = P(x_1) = 0.4$; $\quad \omega_3 = P(x_1)+P(x_2)=0.65$; $\quad \omega_4 = P(x_1)+P(x_2)+P(x_3)=0.8$
$\omega_5 = P(x_1)+P(x_2)+P(x_3)+P(x_4)=0.9$; $\quad \omega_6 = P(x_1)+P(x_2)+P(x_3)+P(x_4)+P(x_5))=0.97$

$c_1 = \omega_1 \times 2 = 0 \times 2 = 0$, $\quad 0 \times 2 = 0$ ☾ 00
$c_2 = \omega_2 \times 2 = 0.4 \times 2 = 0.8$, $\quad 0.8 \times 2 = 1.6$ ☾ 01
$c_3 = \omega_3 \times 2 = 0.65 \times 2 = 1.3$, $\quad 0.3 \times 2 = 0.6$, $\quad 0.6 \times 2 = 1.2$ ☾ 101
$c_4 = \omega_4 \times 2 = 0.8 \times 2 = \underline{1}.6$, $\quad 0.6 \times 2 = \underline{1}.2$, $\quad 0.2 \times 2 = \underline{0.}4$, $\quad 0.4 \times 2 = \underline{0.}8$ ☾ 1100
$c_5 = \omega_5 \times 2 = 0.9 \times 2 = 1.8$, $\quad 0.8 \times 2 = 1.6$, $\quad 0.6 \times 2 = 1.2$, $\quad 0.2 \times 2 = 0.4$ ☾ 1110
$c_6 = \omega_6 \times 2 = 0.97 \times 2 = 1.94$, $\quad 0.94 \times 2 = 1.88$, $\quad 0.88 \times 2 = 1.76$,0 $\quad .76 \times 2 = 1.52$,
$0.52 \times 2 = 1.04$, $\quad 0.04 \times 2 = 0.08$ ☾ 111110

a) Code efficiency $\eta = \dfrac{H(x)}{Lc\ Log2D}$, $H(X) = -\sum_1^6 \mathrm{P(xi)}\ log_2\ P(\mathrm{x}i)$ = 2.1918 bits/message

$$Lc = \bar{l}i = \sum_{i=1}^{6} li\ P(xi) = 2.61\ bits/message$$

$$\rightarrow \eta = \frac{2.1918}{2.61\ Log22} = 83.9\%$$

b) $$P(0) = \frac{\sum_{i=1}^{6} 0i\ P(xi)}{Lc} = \frac{[2\times 0.4 + 1\times 0.25 + 1\times 0.15 + 2\times 0.1 + 1\times 0.07 + 1\times 0.03]}{2.61} = 0.5674$$

$$P(1) = \frac{\sum_{i=1}^{6} 1i\ P(xi)}{Lc}\ or\ P(1) = 1 - P(0) = 0.426$$

HW. Repeat previous example for Ternary Shannon coding ?

Notes: 1. base of log in evaluating li will be 3.

2. The condition of $\boldsymbol{li}$ will be $\left(\frac{1}{3}\right)r$

3. $\boldsymbol{\omega_i}$ is changed into ternary word of length $\boldsymbol{li}$.

Example Develop ternary Shannon code for the following set of messages, $p(x)$ = [0.3 0.2 0.15 0.12 0.1 0.08 0.05]

Solution

$$l_i = \begin{cases} -log3p(xi) \ldots. & ifp(xi) = \left(\frac{1}{3}\right)^r = (\frac{1}{3}, \frac{1}{9}, \frac{1}{27} \ldots) \\ int[-log3p(xi)] + 1 & \ldots.ifp(xi) \neq \left(\frac{1}{3}\right)^r \end{cases}$$

$To\ find\ C5$ then multiply w_5 by 3 L_5 times as below

0.77 × 3= 2.31   2
0.31 × 3= 0.93   0
0.93 × 3=2.79   2

$$\eta = \frac{H(x)}{Lc\ Log2D}$$
$= 73.632\%$

P(0)=0.404

X_i	$P(x_i)$	L_i	ω_i	C_i	0_i
x_1	0.3	2	0	00	2
x_2	0.2	2	0.3	02	1
x_3	0.15	2	0.5	11	0
x_4	0.12	2	0.65	12	0
x_5	0.10	3	0.77	202	1
x_6	0.08	3	0.87	212	0
x_7	0.05	3	0.95	221	0

Procedure for binary coding:

a. Arrange messages in decreasing order of probability.

b. Find out a point in that order in which the sum of probability upward is almost equal to the sum of probability downward. Assign all messages upward as "0" and all messages downward as "1".

c. Repeat the previous step(**b**) many times on upward and downward until all messages are separated.

Example:- Develop Shannon-Fano Code for the following messages: **$P(x_i)$= [0.4 0.25 0.15 0.1 0.07 0.03**] then find:

a. Code efficiency ; b. P(0) & P(1) at the encoder output

x_i	$P(x_i)$	Codeword c_i	L_i	0_i	1_i
x_1	0.4	0	1	1	0
x_2	0.25	10	2	1	1
x_3	0.15	110	3	1	2
x_4	0.1	1110	4	1	3
x_5	0.07	11110	5	1	4
x_6	0.03	11111	5	0	5

$$Lc = 1\times 0.4 + 2\times 0.25 + 3\times 0.15 + 4\times 0.1 + 5\times 0.07 + 5\times 0.03 = 2.25 \text{ bits/message}$$

H(X) = 2.1918 bits/message $\quad -\rightarrow \ \eta = \dfrac{2.1918}{2.25\ Log_2 2} = 97.4\%$

$$P(0) = \frac{\sum_{i=1}^{6} 0i\ P(xi)}{Lc} = 0.4311 \quad \rightarrow P(1) = 0.5689$$

Example:- Develop Fano Code for the following messages: **$P(x_i)$= [0.4 0.2 0.12 0.08 0.08 0.04 0.04 0.04**] then find:
a. Code efficiency ; b. P(0) & P(1) at the encoder output

Solution:-

$Lc = 2.56$ bits/message and

$H(X) = 2.5$ bits/message

$$\rightarrow \eta = \frac{2.5}{2.56} = 97.6\%$$

$$P(0) = \frac{\sum_{i=1}^{6} 0i\, P(xi)}{Lc} = 0.484 \;\; \rightarrow P(1) = 0.515$$

Line	x_i	$P(x_i)$	Codeword c_i	L_i	0_i
	x_1	0.4	0	1	1
1	x_2	0.2	100	3	2
3	x_3	0.12	101	3	1
2	x_4	0.08	1100	4	2
5	x_5	0.08	1101	4	1
4	x_6	0.04	1110	4	1
6	X_7	0.04	11110	5	1
7	X_8	0.04	11111	5	0

Notes:-

1- Number of line sequences represents the order of drawing these lines.

2- For above example, less η is obtained (higher ***Lc***) if starting line is between x_2 and x_3 which gives the same balancing of sum of prob. compared with that used in above solution (0.4↔0.6).

For Ternary Shannon-Fano Code, find out two lines in each step that split the sum of prob. Into almost three equal parts giving them as 0,1,2.

HW:- Develop Ternary Shannon-Fano Code for the following messages: **$P(x_i)$= [0.4 0.25 0.15 0.1 0.07 0.03**] then find:
a. Code efficiency ; b. P(0) , P(1) & P(2) at the encoder output