Classification of codes

Classification of codes:

- Fixed length codes
- Variable length codes
- Distinct codes
- Prefix-free codes
- Uniquely decodable codes
- Instantaneous codes
- Optimal codes

Fixed length codes:
Codeword length is fixed. All symbols are encoded with same number of bits (Code 1, 2 & 7)

Variable length codes:
Codeword length is not fixed (Codes 3, 4, 5, 6 etc)

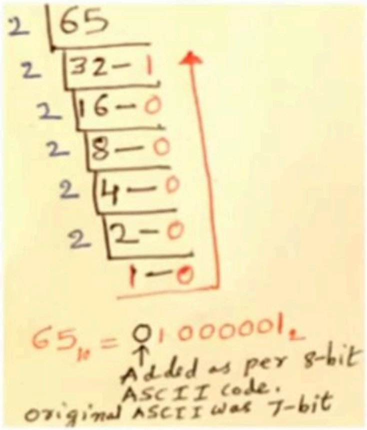
x_i	Code 1	Code 2	Code 3	Code 4	Code 5	Code 6	Code 7	Code 8	…………
x_1	00	11	0	0	0	1	00		
x_2	01	10	1	10	01	01	01		
x_3	10	01	00	110	011	111	00		
x_4	11	00	11	111	0111	0001	11		

Fixed length coding: ASCII (American standard Code for Information Interchange)

Character on Keyboard	ASCII Decimal value	ASCII Binary value
0	48	0011 0000
1	49	0011 0001
2	50	0011 0010
3	51	0011 0011
4	52	0011 0100
5	53	0011 0101
6	54	0011 0110
7	55	0011 0111
8	56	0011 1000
9	57	0011 1001

Character on Keyboard	ASCII Decimal value	ASCII Binary value
A	65	01000001
B	66	01000010
C	67	01000011
D	68	01000100
.		
.		
.		
.		
Y	89	01011001
Z	90	01011010

Character on Keyboard	ASCII Decimal value	ASCII Binary value
a	97	01100001
b	98	01100010
c	99	01100011
d	100	01100100
.		
.		
.		
.		
y	121	01111001
z	122	01111010



Character on Keyboard	ASCII Decimal value	ASCII Binary value
:	58	0011 1010
;	59	0011 1011
<	60	0011 1100
=	61	0011 1101
>	62	0011 1110
?	63	0011 1111
@	64	0100 0000

Variable Length coding:

Different codewords will have different lengths. Examples are Huffman coding, Shannon-Fano, Lempel-Ziv etc.

x_i	Code 1	Code 2	Code 3	Code 4	Code 5	Code 6	Code 7	Code 8	…………
x_1	00	11	0	0	0	1	00		
x_2	01	10	1	10	01	01	01		
x_3	10	01	00	110	011	111	00		
x_4	11	00	11	111	0111	0001	11		

Distinct codes:

Each codeword is distinguishable from other codewords (Codes 1, 2)

In code 7, x_1, and x_3 have same codeword. So, it is not distinct code

Prefix-free codes:

No codeword should be prefix to other codeword (Codes 1, 2, & 4)

Observe code 3 (which is not prefix-free):

x_1	0	
x_2	1	
x_3	00	x_1x_1 or x_3 --- confusion while decoding
x_4	11	x_2x_2 or x_4 --- confusion while decoding

Uniquely decodable codes:

All prefix-free codes are uniquely decodable. A sufficient condition to ensure that no codeword is prefix of another. Hence prefix-free codes are uniquely decodable codes.

Note: prefix-free condition is not a necessary condition for unique decodability.

Example:

x_1	0	
x_2	01	These are not prefix free. It is uniquely decodable.
x_3	011	
x_4	0111	

For example, decode sequence 011101 --- x_4x_2. This decoding is OK.

Another example, 0111 ----- x4 only. OK
Or $x_1$11 Or $x_2$11 Or $x_3$1 ⇒ These are all meaning less way of decoding

Instantaneous codes (Code 1, 2, & 4):
Prefix-free codes sometimes known as instantaneous codes. No codeword should be prefix of another codeword.

x_i	Code 1	Code 2	Code 3	Code 4	Code 5	Code 6	Code 7	Code 8	…………
x_1	00	11	0	0	0	1	00		
x_2	01	10	1	10	01	01	01		
x_3	10	01	00	110	011	111	00		
x_4	11	00	11	111	0111	0001	11		

Optimal codes (Code 1, 2, & 4):
A code is said to be optimal if it is instantaneous and has minimum L for a given source. Huffman codes are optimum codes.

Krafts inequality

Kraft's inequality: A necessary and sufficient condition for the existance of a binary code with codewords having lengths $n_1 \le n_2 \le n_3 \le n_4 \le \dots \le n_L$ that satisfy the prefix condition is

$$\sum_{K=1}^{L} 2^{-n_k} \le 1$$

Kraft's inequality: A necessary and sufficient condition for the existance of a binary code with codewords having lengths $n_1 \le n_2 \le n_3 \le n_4 \le \dots \le n_L$ that satisfy the prefix condition is

$$\sum_{K=1}^{L} 2^{-n_k} \le 1$$

A, B, C, D

$$2^{-1} + 2^{-2} + 2^{-3} + 2^{-3}$$

$$= 0.5 + 0.25 + 0.125 + 0.125$$

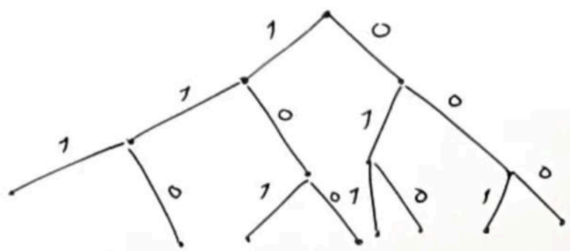
$$1.0$$

A – 0

B - 10

C - 110

D - 111



Instantaneous codes

Source coding

- → prefix codes / Instantaneous codes
- → Shannon Fano codes
- → Huffman codes
- → extended Huffman codes.

Instantaneous codes / prefix codes

A prefix code, in which no code is a prefix another code.

Instantaneous codes

Source coding

- → prefix codes / Instantaneous codes
- → Shannon Fano codes
- → Huffman codes
- → Extended Huffman codes.

Instantaneous codes / prefix codes

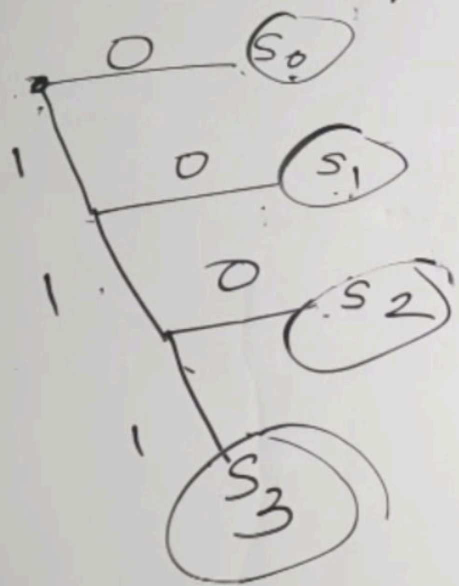
A prefix code, is which no code is a prefix another code.

→ 0

→ 01

	Code I	Code II	Code III
S_0	0	0	0
S_1	1	10	01
S_2	00	110	011
S_3	11	111	0111
	✗ prefix	✓ prefix code	✗ prefix code

Decision tree



$s_0 \rightarrow 0$

$s_1 \rightarrow 10$

$s_2 \rightarrow 110$

$s_3 \rightarrow 111$

Construction of Prefix codes

$S = \{s_1, s_2, s_3, s_4, s_5\}$

$X = \{0, 1\}$

$s_1 \longrightarrow 0$ ✓

$s_2 \longrightarrow 1$

$s_3 \longrightarrow 1$

$s_4 \longrightarrow 1$

$s_5 \longrightarrow 1$

Construction of Prefix codes

$S = \{s_1, s_2, s_3, s_4, s_5\}$

$X = \{0, 1\}$

$s_1 \longrightarrow 0$

$s_2 \longrightarrow 10$

$s_3 \longrightarrow 110$

$s_4 \longrightarrow 1110$

$s_5 \longrightarrow 1111$

Construction of Prefix codes

$S = \{s_1, s_2, s_3, s_4, s_5\}$

$X = \{0, 1\}$

$s_1 \rightarrow 0$ ✓

$s_2 \rightarrow 10$ ✓

$s_3 \rightarrow 110$ ✓

$s_4 \rightarrow 1110$ ✓

$s_5 \rightarrow 1111$ ✓

$s_1 \rightarrow 00$

$s_2 \rightarrow 01$

$s_3 \rightarrow 10$

$s_4 \rightarrow$

$s_5 \rightarrow$

Construction of Prefix codes

$S = \{s_1, s_2, s_3, s_4, s_5\}$

$X = \{0, 1\}$

$s_1 \rightarrow 0$ ✓	$s_1 \rightarrow 00$
$s_2 \rightarrow 10$ ✓	$s_2 \rightarrow 01$
$s_3 \rightarrow 110$ ✓	$s_3 \rightarrow 10$
$s_4 \rightarrow 1110$ ✓	$s_4 \rightarrow 11$
$s_5 \rightarrow 1111$ ✓	$s_5 \rightarrow 11$

Construction of Prefix codes

$S = \{s_1, s_2, s_3, s_4, s_5\}$

$X = \{0, 1\}$

$s_1 \longrightarrow 0$

$s_2 \longrightarrow 10$

$s_3 \longrightarrow 110$

$s_4 \longrightarrow 1110$

$s_5 \longrightarrow 1111$

$s_1 \longrightarrow 00$

$s_2 \longrightarrow 01$

$s_3 \longrightarrow 10$

$s_4 \longrightarrow 110$

$s_5 \longrightarrow 111$

Construction of Prefix codes

$S = \{s_1, s_2, s_3, s_4, s_5\}$

$X = \{0, 1\}$

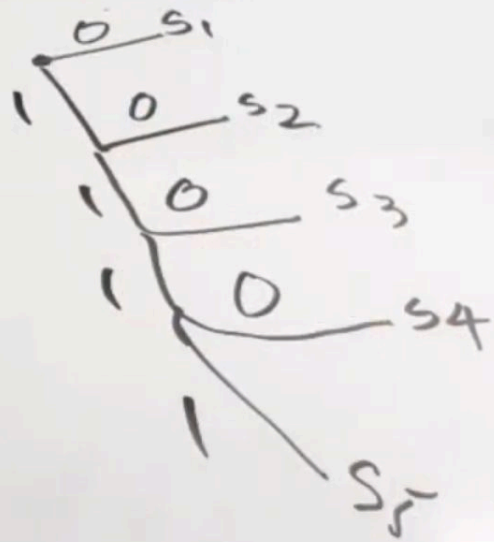
$s_1 \rightarrow 0$ ✓

$s_2 \rightarrow 10$ ✓

$s_3 \rightarrow 110$ ✓

$s_4 \rightarrow 1110$ ✓

$s_5 \rightarrow 1111$ ✓



Kraft McMillan Inequality

Source Message $\{s_0, s_1, s_2 \cdots s_{K-1}\}$ — S_K

Probabilities $\{P_0, P_1 \cdots P_{K-1}\}$

$S_K \longrightarrow$ lengths l_K. $(K \rightarrow 0 - K-1)$

then

$$\left[\sum_{K=0}^{K-1} 2^{-l_K} \le 1 \right]$$

$\{0, 10, 110, \cdots\}$

$1 \quad 2 \quad 3$

$2^{-1} + 2^{-2} + 2^{-3} \cdots \le 1$

s_0 – 0 (1)

s_1 – 10 (2)

s_2 – 110 (3)

s_3 – 1110 (4)

s_4 – 1111 (4)

$\rightarrow S_K$

$$\sum_{K=0}^{K-1} 2^{-l_K} \leq 1$$

$$2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} + 2^{-4} = 1$$

Source coding theorem

Source Coding Theorem : Suppose a DMS outputs a symbol every t seconds. Each symbol is selected from a finite set of symbols x_i, $i = 1, 2, ----, L$, occuring with probabilities $P(x_i)$, $i = 1, 2, ---, L$. The entropy of this DMS in bits per source symbol is

$$H(X) = \sum_{i=1}^{L} P(x_i) \cdot \log \frac{1}{P(x_i)} \leq \log_2 L$$

The equality holds when the symbols are equally likely.

Source Coding Theorem : Suppose a DMS outputs a symbol every t seconds. Each symbol is selected from a finite set of symbols x_i, $i = 1, 2, \dots, L$, occuring with probabilities $P(x_i)$, $i = 1, 2, \dots, L$. The entropy of this DMS in bits per source symbol is

$$H(X) = \sum_{i=1}^{L} P(x_i) \cdot \log \frac{1}{P(x_i)} \leq \log_2 L$$

The equality holds when the symbols are equally likely.

$$H(X) = -\sum_{i=1}^{L} P(x_i) \log P(x_i) \leq \log_2 L$$

x_1 $\quad P(x_1) = 0.5$

x_2 $\quad P(x_2) = 0.5$

$$\begin{aligned}
H(x) &= -0.5 \log_2 0.5 - 0.5 \log_2 0.5 \\
&= -0.5 \log_2 2^{-1} - 0.5 \log_2 2^{-1} \\
&= -0.5(-1) \log_2 2 - (0.5)(-1) \log_2 2 \\
&= 0.5 + 0.5 = 1
\end{aligned}$$

Coding Theorem : Suppose a DMS outputs a symbol every t seconds. Each symbol is selected from a finite set of symbols x_i, $i = 1, 2, \ldots, L$, occuring with probabilities $P(x_i)$, $i = 1, 2, \ldots, L$. The entropy of this DMS in bits per source symbol is

$$H(X) = \sum_{i=1}^{L} P(x_i) \cdot \log \frac{1}{P(x_i)} \leq \log_2 L$$

The equality holds when the symbols are equally likely.

$$H(X) = -\sum_{i=1}^{L} P(x_i) \log P(x_i) \leq \log_2 L$$

x_1 $\quad P(x_1) = 0.5$

x_2 $\quad P(x_2) = 0.5$

$$H(x) = -0.5 \log_2 0.5 - 0.5 \log_2 0.5$$
$$= -0.5 \log_2 2^{-1} - 0.5 \log_2 2^{-1}$$
$$= -0.5(-1) \log_2 2 - (0.5)(-1) \log_2 2$$
$$= 0.5 + 0.5 = 1$$

$$3010 \overline{)\,4771\,} \; 1.58$$
$$3010$$

x_1 $\quad P(x_1) = 0.5$

x_2 $\quad P(x_2) = 0.25$

x_3 $\quad P(x_3) = 0.25$

LHS

$$H(x) = -0.5 \log_2 0.5 - 0.25 \log_2 0.25 - 0.25 \log_2 0.25$$
$$= -0.5 \log_2 2^{-1} - 0.25 \log_2 2^{-2} - 0.25 \log_2 2^{-2}$$
$$= +0.5 + 0.5 + 0.5 = 1.5$$

RHS

$$\log_2 3 = \frac{\log_{10} 3}{\log_{10} 2} = \frac{0.4771}{0.3010} = 1.58$$

Fixed length code:

$x_i, \ i = 1, 2, ----, L$

$$R = \lceil \log_2 L \rceil$$

or

$$R = \lfloor \log_2 L \rfloor + 1 \quad \text{if } L \text{ is not power of } 2.$$

$$= \log_2 L \quad \text{if } L \text{ is power of } 2.$$

A, B, ----, Z

$$R = \lceil \log_2 26 \rceil = \lceil 4.6 \rceil = 5$$

A, B, C, D, E, F, G, H

$$R = \log_2 8 = \log_2 2^3 = 3$$

Shannon Fano coding

Shannon Fano Encoding Algorithm

1. The messages are first written in the order of decreasing probability.
2. Then divide the messages set into two most equiprobable subset X and Y.
3. The message of 1st set X is given bit 0 and message in the 2nd subset is given bit 1.
4. The procedure is now applied for each set seprately till end.
5. Finally we get the code word for respective symbol.
6. Calculation

→ efficiency $(\eta) = \frac{H}{\bar{H}}$

where, H = Entropy $= \sum_{i=1}^{n} p_i \log_2 ($

probability.

2. Then divide the messages set into two most equiprobable subset X and Y.
3. The message of 1st set X is given bit 0 and message in the 2nd subset is given bit 1.
4. The procedure is now applied for each set seprately till end.
5. Finally we get the code word for respective symbol.
6. Calculation

→ efficiency $(\eta) = \frac{H}{\hat{H}}$

where, $H = \text{Entropy} = \sum_{i=1}^{n} P_i \log_2\left(\frac{1}{P_i}\right)$

$\hat{H} = \sum_{i=1}^{n} P_i q_i$

→ Redundancy

$R_e = 1 - \eta$

Shannon - Fano encoding Algorithm

Example - Find the code words occuring in the probability $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$ for symbols s_1, s_2, s_3 & s_4. Find efficiency and redundancy of code.

Example - Find the code words occuring in the probability {$\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{8}$} for symbols s_1, s_2, s_3 & s_4. Find efficiency and redundancy of code.

Symbol	Prob			
s_1	$1/2$	0		
s_2	$1/4$		0	
s_3	$1/8$	1	1	0
s_4	$1/8$			1

Example - Find the code words occuring in the probability $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$ for symbols s_1, s_2, s_3 & s_4. Find efficiency and redundancy of code.

Symbol	Prob	Codeword	Length
s_1	$1/2$	0	1
s_2	$1/4$	10	2
s_3	$1/8$	110	3
s_4	$1/8$	111	3

s_3	$1/8$	$1 \quad 0$	110	3
s_4	$1/8$	$1 \quad 1$	111	3

- Efficiency $\eta = \dfrac{H}{\hat{H}}$

where H = entropy $= \sum_{i=1}^{4} p_i \log_2\left(\frac{1}{p_i}\right)$

$$= \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \left(\frac{1}{8}\log_2 8\right) \times 2$$

$$= \frac{1}{2} + \frac{1}{2} + \frac{3}{4}$$

$$= 1.75 \text{ bits/symbol}$$

$$= \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \left(\frac{1}{8}\log_2 8\right) \times 2$$

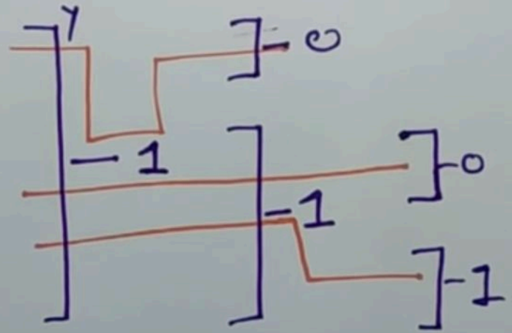
$$= \frac{1}{2} + \frac{1}{2} + \frac{3}{4}$$

$$= 1.75 \text{ bits / symbol}$$

$$\hat{H} = \sum_{i=1}^{4} p_i n_i$$

$$= \frac{1}{2} \times 1 + 1/4 \times 2 + \frac{1}{8} \times 3 \times 2$$

$$= 1.75 \text{ bits / symbol}$$

s_2	$1/4$		10	2
s_3	$1/8$		110	3
s_4	$1/8$		111	3

- Efficiency $\eta = \frac{H}{\hat{H}} = \frac{1.75}{1.75} = 1$

where H = entropy $= \sum_{i=1}^{4} p_i \log_2\left(\frac{1}{p_i}\right)$

$$= \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \left(\frac{1}{8}\log_2 8\right) \times 2$$

$$= \frac{1}{2} + \frac{1}{2} + \frac{3}{4}$$

$$\left(\frac{1}{8}\log_2 8\right)\times 2$$

$$= \frac{1}{2} + \frac{1}{2} + \frac{3}{4}$$

$$= 1.75 \text{ bits / symbol}$$

$$\hat{H} = \sum_{i=1}^{4} p_i n_i$$

$$= \frac{1}{2}\times 1 + 1/4 \times 2 + \frac{1}{8}\times 3 \times 2$$

$$= 1.75 \text{ bits / symbol}$$

- Redundancy $R_e = 1 - \eta$

$$= 1 - 1 = 0$$

Example Find the code word for the probability

$\frac{1}{4}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{16}, \frac{1}{16}, \frac{1}{16}, \frac{1}{16}$ for symbols $S_1, S_2, \dots S_8$.

Find the code efficiency and redundancy.

Symbol Prob.

$\frac{1}{4}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{16}, \frac{1}{16}, \frac{1}{16}, \frac{1}{16}$ for symbols $S_1, S_2, \dots S_8$.

Find the code efficiency and Redundancy.

Symbol	Prob.				
S_1	1/4	0	0		
S_2	1/4		1		
S_3	1/8		0	0	
S_4	1/8			1	
S_5	1/16	1		0	0
S_6	1/16		1		1
S_7	1/16			1	0
S_8	1/16				1

Symbol	Prob.	Codeword	length
s_1	1/4	00	2
s_2	1/4	01	2
s_3	1/8	100	3
s_4	1/8	101	3
s_5	1/16	1100	4
s_6	1/16	1101	4
s_7	1/16	1110	4
s_8	1/16	1110	4

– H

- $H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$

$= 2\left(\frac{1}{4}\log_2(4)\right) + 2\left(\frac{1}{8}\log_2(8)\right) + 4\left(\frac{1}{16}\log_2 16\right)$

$= 1 + \frac{3}{4} + 1$

$= 2.75$ bits / symbol

- $\hat{H} = \sum p_i n_i$

$= 2\left(\frac{1}{4}\times 2\right) + 2\left(\frac{1}{8}\times 3\right) + 4\left(\frac{1}{16}\times 4\right)$

$= 2.75$ bits / symbol

$$= 2\left(\frac{1}{4}\log_2(4)\right) + 2\left(\frac{1}{8}\log_2(8)\right) + 4\left(\frac{1}{16}\log_2 16\right)$$

$$= 1 + \frac{3}{4} + 1$$

$$= 2.75 \text{ bits / Symbol}$$

- $\hat{H} = \sum p_i n_i$

$$= 2\left(\frac{1}{4}\times 2\right) + 2\left(\frac{1}{8}\times 3\right) + 4\left(\frac{1}{16}\times 4\right)$$

$$= 2.75 \text{ bits / Symbol}$$

- efficiency $\eta = \frac{H}{\hat{H}} = \frac{2.75}{2.75} = 1$

- Redundancy $R_c = 1 - \eta = 1 - 1 = 0$

Shannon - Fano encoding Algorithm [problem on Ambiguity]

Example - Apply the sannon - Fano coding procedure for the message $X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]$

Probability $P = [0.4, 0.2, 0.12, 0.08, 0.08, 0.08, 0.04]$

message	Prob
x_1	0.4
x_2	0.2
x_3	0.12
x_4	0.08
x_5	0.08
x_6	0.08
x_7	0.04

$\{x_1\}$ $\{x_2, x_3, \dots x_7\}$

0.4 0.6

$\{x_1, x_2\}$ $\{x_3, x_4, \dots x_7\}$

0.6 0.4

message	Prob	Codeword	length
x_1	0.4	0	1
x_2	0.2	100	3
x_3	0.12	101	3
x_4	0.08	1100	4
x_5	0.08	1101	4
x_6	0.08	1110	4
x_7	0.04	1111	4

$\{x_1\}$ $\{x_2, x_3, \ldots x_7\}$

0.4   0.6

$\{x_1, x_2\}$ $\{x_3, x_4, \ldots x_7\}$

0.6   0.4

- $\{x_1\}$ $\{x_2, x_3, \dots x_7\}$

  0.4 0.6

- $\{x_1, x_2\}$ $\{x_3, x_4, \dots x_7\}$

  0.6 0.4

message	Prob	codeword	length
x_1	0.4	00	2
x_2	0.2	01	2
x_3	0.12	100	3
x_4	0.08	101	3
x_5	0.08	110	2
x_6	0.08	1110	4
x_7	0.04	1111	4

message	Prob	Codeword	length
x_1	0.4	0	1
x_2	0.2	100	3
x_3	0.12	101	3
x_4	0.08	1100	4
x_5	0.08	1101	4
x_6	0.08	1110	4
x_7	0.04	1111	4

- $\{x_1\}$ $\{x_2, x_3, \ldots x_7\}$
  0.4   0.6
- $\{x_1, x_2\}$ $\{x_3, x_4, \ldots x_7\}$
  0.6   0.4

$$\eta = \frac{H}{\hat{H}}$$

$$\hat{H} = \sum p_i n_i$$

$$= 0.4 \times 1 + 0.2 \times 3 + 0.12 \times 3 + 3 \times 0.08 \times 4 + 0.04 \times 4$$

$$= 2.48 \text{ bits/symbol}$$

message	Prob	codeword	length
x_1	0.4	00	2
x_2	0.2	01	2
x_3	0.12	100	3
x_4	0.08	101	3
x_5	0.08	110	3
x_6	0.08	1110	4
x_7	0.04	1111	4

$$\hat{H} = \sum p_i n_i$$

$$= 0.4 \times 2 + 0.2 \times 2 + 0.12 \times 3 + 0.08 \times 3 \times 2 + 0.08 \times 4 + 0.04 \times 4$$

$$= 2.52 \text{ bits/symbol}$$

[0.4 , 0.2 , 0.12 , 0.08 , 0.08 , 0.08 , 0.04]

message	Prob	codeword	length
x_1	0.4	0	1
x_2	0.2	100	3
x_3	0.12	101	3
x_4	0.08	1100	4
x_5	0.08	1101	4
x_6	0.08	1110	4
x_7	0.04	1111	4

- $\{x_1\}\ \{x_2, x_3, \ldots x_7\}$

  0.4   0.6

- $\{x_1, x_2\}\ \{x_3, x_4, \ldots x_7\}$

  0.6   0.4

$$\eta = \frac{H}{\hat{H}}$$

$$\hat{H} = \sum p_i n_i$$

$$= 0.4\times1 + 0.2\times3 + 0.12\times3 + 3\times0.08\times4 + 0.04\times4$$

$$= 2.48 \text{ bits/symbol} \checkmark$$

message	Prob	codeword	length
x_1	0.4	00	2
x_2	0.2	01	2
x_3	0.12	100	3
x_4	0.08	101	3
x_5	0.08	110	3
x_6	0.08	1110	4
x_7	0.04	1111	4

$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$

$$= 0.4 \log_2\left(\frac{1}{0.4}\right) + 0.2 \log_2\left(\frac{1}{0.2}\right) + 0.12 \log_2\left(\frac{1}{0.12}\right) + 3 \times 0.08 \log_2\left(\frac{1}{0.08}\right) + 0.04 \log_2\left(\frac{1}{0.04}\right) = 2.42 \text{ bits/symbol}$$

$$\hat{H} = \sum p_i n_i$$

$$= 0.4 \times 2 + 0.2 \times 2 + 0.12 \times 3 + 0.08 \times 3 \times 2 + 0.08 \times 4 + 0.04 \times$$

$$= 2.52 \text{ bits/symbol} \quad X$$

x_5	0.08	1101	4
x_6	0.08	1110	4
x_7	0.04	1111	4

- $\{x_1\}\ \{x_2, x_3, \ldots x_7\}$
  0.4 0.6
- $\{x_1, x_2\}\ \{x_3, x_4, \ldots x_7\}$
  0.6 0.4

$$\boxed{\eta = \frac{H}{\hat{H}}}$$

$$= \frac{2.42}{2.48}$$

$$= \boxed{97.6\%}$$

$\hat{H} = \Sigma p_i n_i$

$= 0.4\times1 + 0.2\times3 + 0.12\times3 +$
$3\times0.08\times4 + 0.04\times4$

$= 2.48$ bits/symbol ✓

message	Prob	Codeword	length
x_1	0.4	00	2
x_2	0.2		2
x_3	0.12		
x_4	0.08		

Huffman Coding

Huffman Coding

1. The source symbols are arranged in order of decreasing prob. Then the two of lowest probability are assigned bit 0 and 1.
2. Then combine last two symbols and move the combined Symbol as high as possible.
3. Repeat the above step untill end.
4. Code for each symbol is found by moving backward.
5. Calculation

efficiency $\eta = \dfrac{H}{L \log_2 r}$

L = Avg Codeword $= \sum_{i=1}^{n} p_i n_i$

$H = \sum^{n} p_i \log_2 \left(\frac{1}{p_i}\right)$

Then the two of lowest probability are assigned bit 0 and 1.

2. Then combine last two symbols and move the combined Symbol as high as possible.
3. Repeat the above step untill end.
4. Code for each symbol is found by moving backward.
5. Calculation

efficiency $\eta = \frac{H}{L \log_2 r}$

L = Avg Codeword $= \sum_{i=1}^{n} p_i n_i$

$H = \sum_{i=1}^{n} p_i \log_2\left(\frac{1}{p_i}\right)$

Varrimee $\sigma^2 = \sum_{i=1}^{n} p_i (n_i - L)^2$

Then the two of lowest probability are assigned bit 0 and 1.

2. Then combine last two symbols and move the combined Symbol as high as possible.
3. Repeat the above step untill end.
4. Code for each symbol is found by moving backward.
5. Calculation

efficiency $\eta = \dfrac{H}{L \log_2 r}$

binary $r = 2$ (0, 1)

ternary $r = 3$ (0, 1, 2)

Quaternary $r = 4$ (0, 1, 2, 3)

L = Avg Codeword $= \sum_{i=1}^{n} p_i n_i$

$H = \sum_{i=1}^{n} p_i \log_2\left(\frac{1}{p_i}\right)$

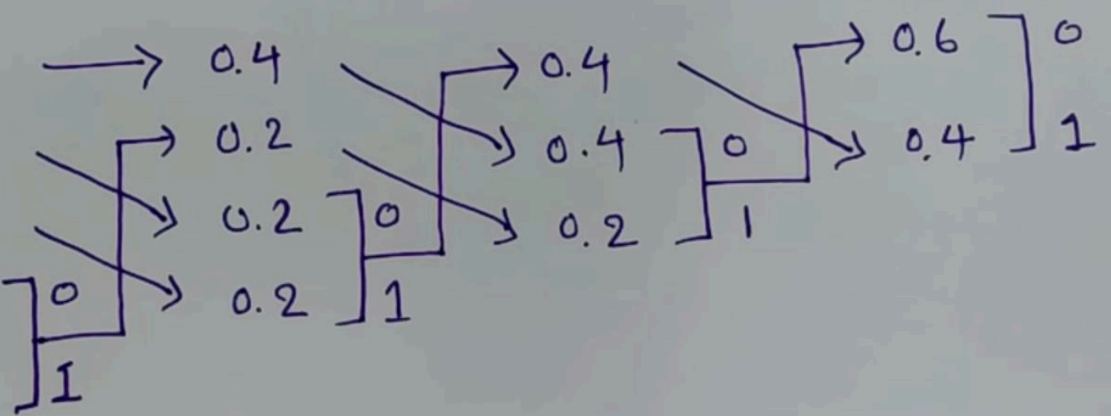
Variance $\sigma^2 = \sum_{i=1}^{n} p_i (n_i - L)^2$

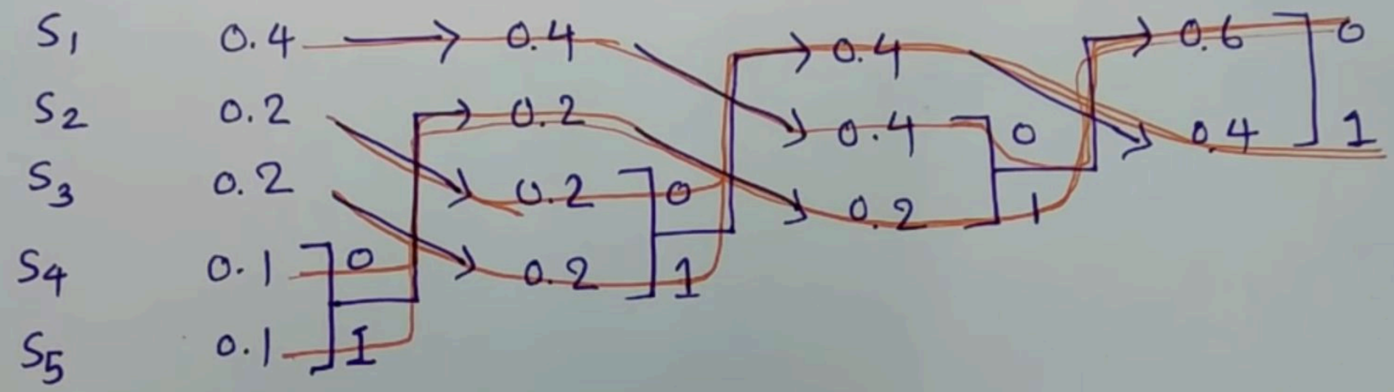
Huffman Coding Algorithm

Example : Alphabet with prob = $\{0.4, 0.2, 0.2, 0.1, 0.1\}$ For symbols $\{s_1, s_2, \ldots, s_5\}$. Find Huffman codes and also Find efficiency & variance

<u>Example</u> : Alphabet with prob = {0.4, 0.2, 0.2, 0.1, 0.1}
For Symbols $\{s_1, s_2, \dots, s_5\}$. Find Huffman codes and also Find efficiency & variance

Symbol	Prob
S_1	0.4
S_2	0.2
S_3	0.2
S_4	0.1
S_5	0.1





Symbol	Codeword	length
S_1	00	2
S_2	10	2
S_3	11	2
S_4	010	3
S_5	011	3

Symbol	Codeword	length
s_1	00	2
s_2	10	2
s_3	11	2
s_4	010	3
s_5	011	3

- Entropy

$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$

$$= 0.4 \log_2\left(\frac{1}{0.4}\right) + 2 \times 0.2 \log_2\left(\frac{1}{0.2}\right) + 2 \times 0.1 \log_2\left(\frac{1}{0.1}\right)$$

$$= 2.1216 \text{ bits/symbol}$$

- $L = \sum p_i n_i$

$=$

S_5	011	3

$$2 \times 0.2 \log_2\left(\frac{1}{0.2}\right) +$$
$$2 \times 0.1 \log_2\left(\frac{1}{0.1}\right)$$
$$= 2.1216 \text{ bits / symbol}$$

– $L = \sum p_i n_i$

$$= 2 \times 0.4 + 2(2 \times 0.2) + 3 \times 0.1 \times 2$$

$$= 2.2 \text{ bits / symbol}$$

– $\eta = \dfrac{H}{L \log_2 r} = \dfrac{2.1216}{2.2 \times \log_2 2} = 96.4\,\%$

– $\sigma^2 = \sum p_i (n_i - L)^2$

$$= 0.4\,(2-2.2)^2 + 2 \times 0.2\,(2-2.2)^2 + 2 \times 0.1\,(3-2.2)^2$$

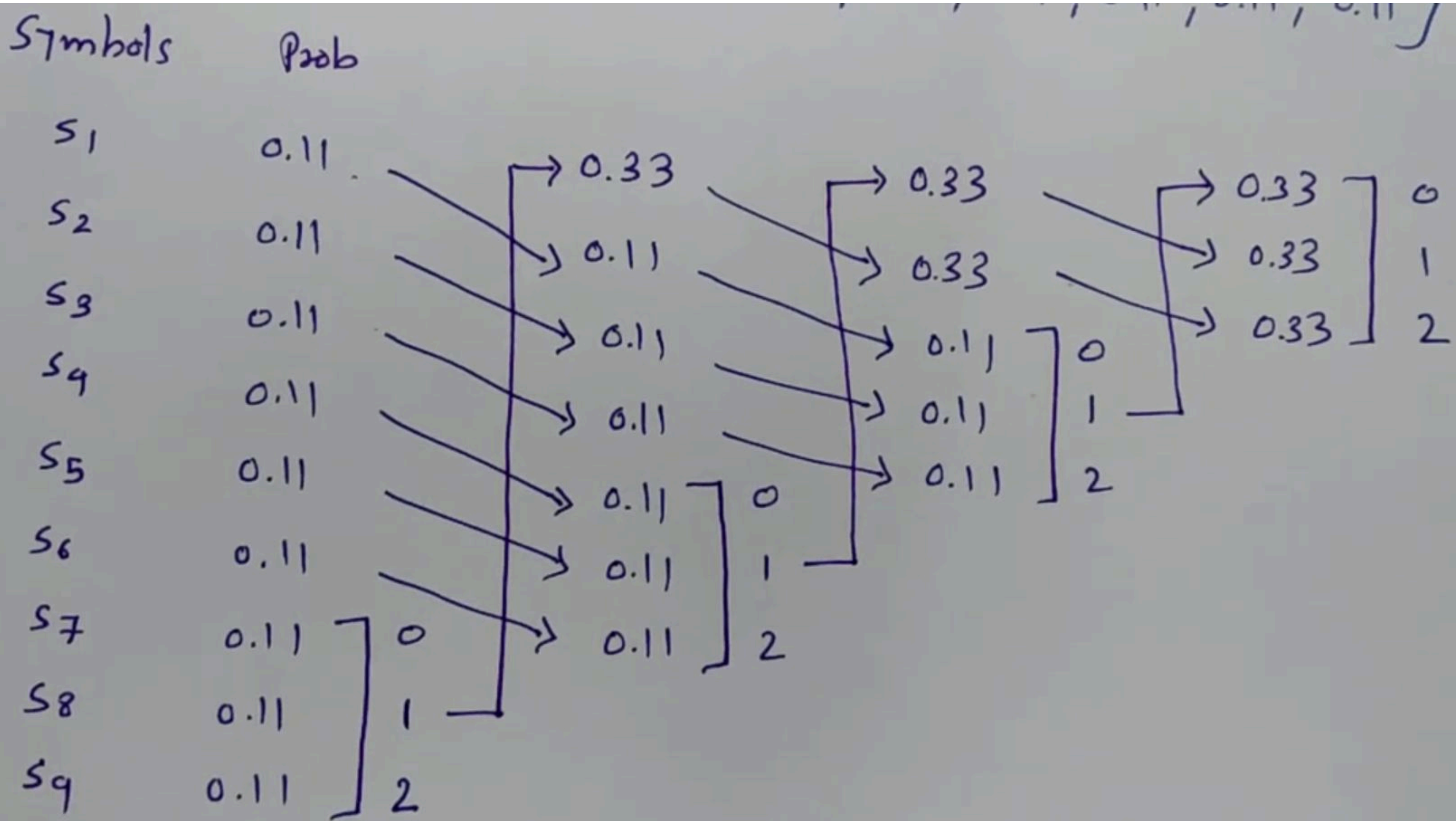
$$= 0.16$$

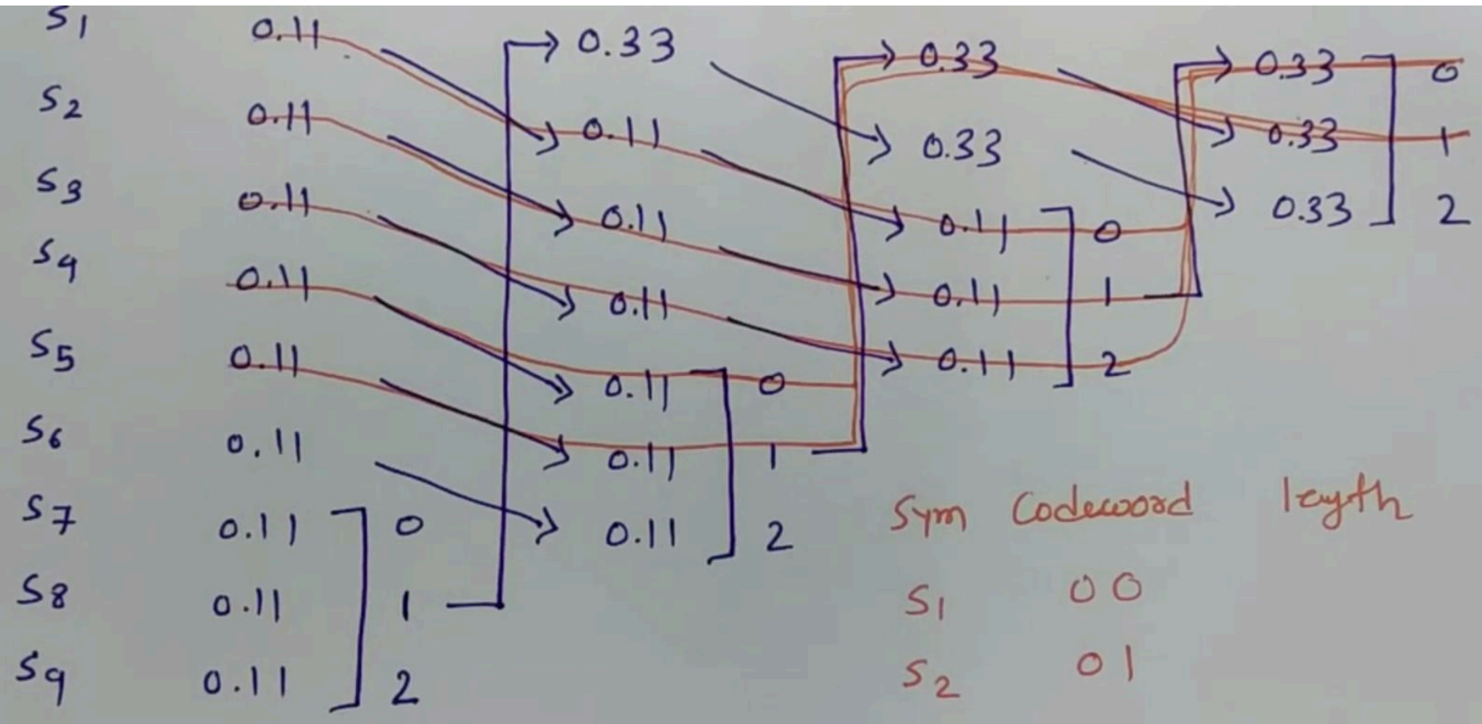
↑ As Low as possible.

Huffman coding procedure and example [ternary code].

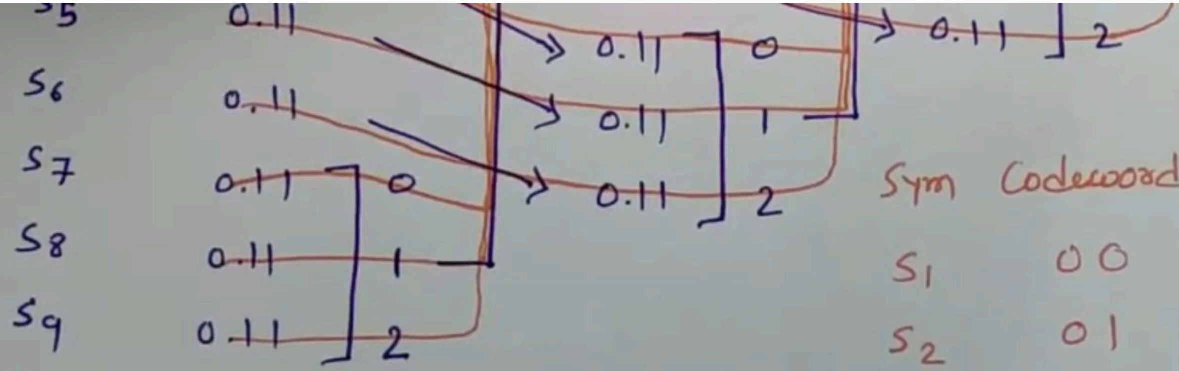
- A source produces 9 symbols $(s_1, s_2, \ldots, s_9)$. construct bin ternary & Quaternary Huffman coding by moving symbols as high as possible. Also find efficiency & variance of the coding. Probability of symbols is given by

$$P_i = \{0.11, 0.11, 0.11, 0.11, 0.11, 0.11, 0.11, 0.11, 0.11\}$$

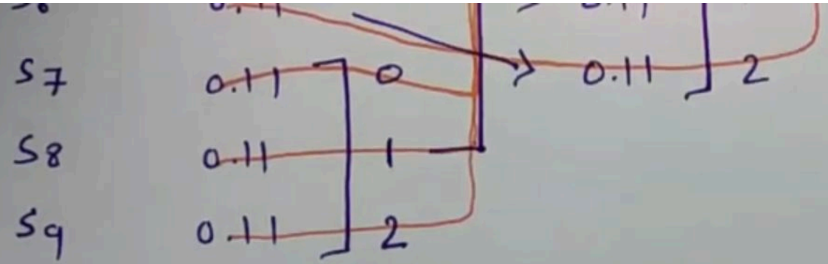

Symbols
Prob
s_1 0.11
s_2 0.11
s_3 0.11
s_4 0.11
s_5 0.11
s_6 0.11
s_7 0.11 0
s_8 0.11 1
s_9 0.11 2
0.33
0.11
0.11
0.11
0.11 0
0.11 1
0.11 2
0.33
0.33
0.11 0
0.11 1
0.11 2
0.33 0
0.33 1
0.33 2



Sym	Codeword	length
s_1	00	
s_2	01	
s_3	02	
s_4	10	
s_5	11	
s_6		



Sym	Codeword	length
s_1	00	2
s_2	01	2
s_3	02	2
s_4	10	2
s_5	11	2
s_6	12	2
s_7	20	2
s_8	21	2
s_9	22	2



$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$

$$= 9\left(0.11 \log_2\left(\frac{1}{0.11}\right)\right)$$

$$= 3.1525 \text{ bits/symbol}$$

$$L = \sum p_i n_i$$

$$= 9\,(0.11 \times 2)$$

$$= 1.98 \text{ bits/symbol}$$

Sym	Codeword	length
s_1	00	2
s_2	01	2
s_3	02	2
s_4	10	2
s_5	11	2
s_6	12	2
s_7	20	2
s_8	21	2
s_9	22	2

$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$

$$= 9\left(0.11 \log_2\left(\frac{1}{0.11}\right)\right)$$

$$= 3.1525 \text{ bits/symbol}$$

$$L = \sum p_i n_i$$

$$= 9(0.11 \times 2)$$

$$= 1.98 \text{ bits/symbol}$$

$$\eta = \frac{H}{L \log_2 r}$$

$$= \frac{3.1525}{1.98 \times \log_2 3} = \frac{3.1525 \times \log 2}{1.98 \times \log 3} = 100\%$$

s_3	02	2
s_4	10	2
s_5	11	2
s_6	12	2
s_7	20	2
s_8	21	2
s_9	22	2

$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$

$$= 9\left(0.11 \log_2\left(\frac{1}{0.11}\right)\right)$$

$$= 3.1525 \text{ bits/symbol}$$

$$L = \sum p_i n_i$$

$$= 9(0.11 \times 2)$$

$$= 1.98 \text{ bits/symbol}$$

$$\eta = \frac{H}{L \log_2 r}$$

$$= \frac{3.1525}{1.98 \times \log_2 3} = \frac{3.1525 \times \lg 2}{1.98 \times \lg 3} = 100\%$$

s_3	02	2
s_4	10	2
s_5	11	2
s_6	12	2
s_7	20	2
s_8	21	2
s_9	22	2

$$\sigma^2 = \sum p_i (n_i - L)^2$$

$$= 9 \times 0.11 \times (1.98 - 2)^2$$

$$=$$

Huffman Coding procedure and example [Quaternary Code]

mbols	Probability	
s_1	0.11	
s_2	0.11	
s_3	0.11	
s_4	0.11	
s_5	0.11	
s_6	0.11	0
s_7	0.11	1
s_8	0.11	2
s_9	0.11	3

0.44	
0.11	
0.11	0
0.11	1
0.11	2
0.11	3

0.44	0
0.44	1
0.11	2
↑ missing	3

→ Add Symbol with 0 prob.

Symbols	Probability	C	ni
s_1	0.11	2	1
s_2	0.11	3	1
s_3	0.11	00	2
s_4	0.11	01	2
s_5	0.11	02	2
s_6	0.11	03	2
s_7	0.11	10	2
s_8	0.11	11	2
s_9	0.11	12	2
s_{10}	0	13	2

Stage 1: 0.11, 0.11, 0.11, 0 → 0, 1, 2, 3

Stage 2: 0.33, 0.11, 0.11, 0.11, 0.11, 0.11, 0.11 (last four → 0, 1, 2, 3)

Stage 3: 0.44, 0.33, 0.11, 0.11 → 0, 1, 2, 3

$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$

$$= 9\left(0.11 \log_2\left(\frac{1}{0.11}\right)\right)$$

$$= 3.15 \text{ bits / symbol.}$$

s_5	0.11		0.11	1	02	2
s_6	0.11		0.11	2	03	2
s_7	0.11	0	0.11	3	10	2
s_8	0.11	1			11	2
s_9	0.11	2			12	2
s_{10}	0	3			13	2

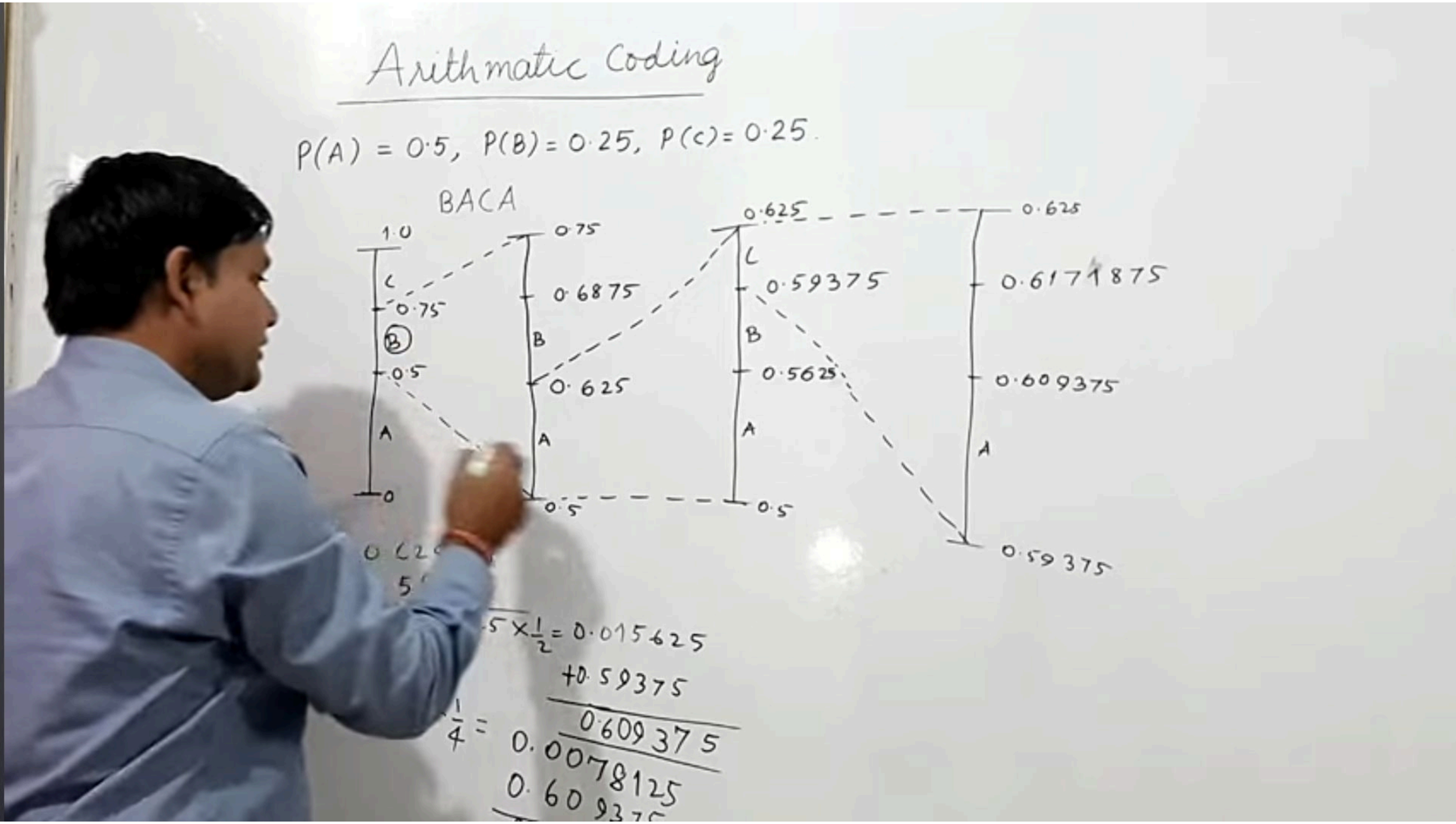
$$H = \sum p_i \log_2\left(\frac{1}{p_i}\right)$$
$$= 9\left(0.11 \log_2\left(\frac{1}{0.11}\right)\right)$$
$$= 3.15 \text{ bits/symbol.}$$

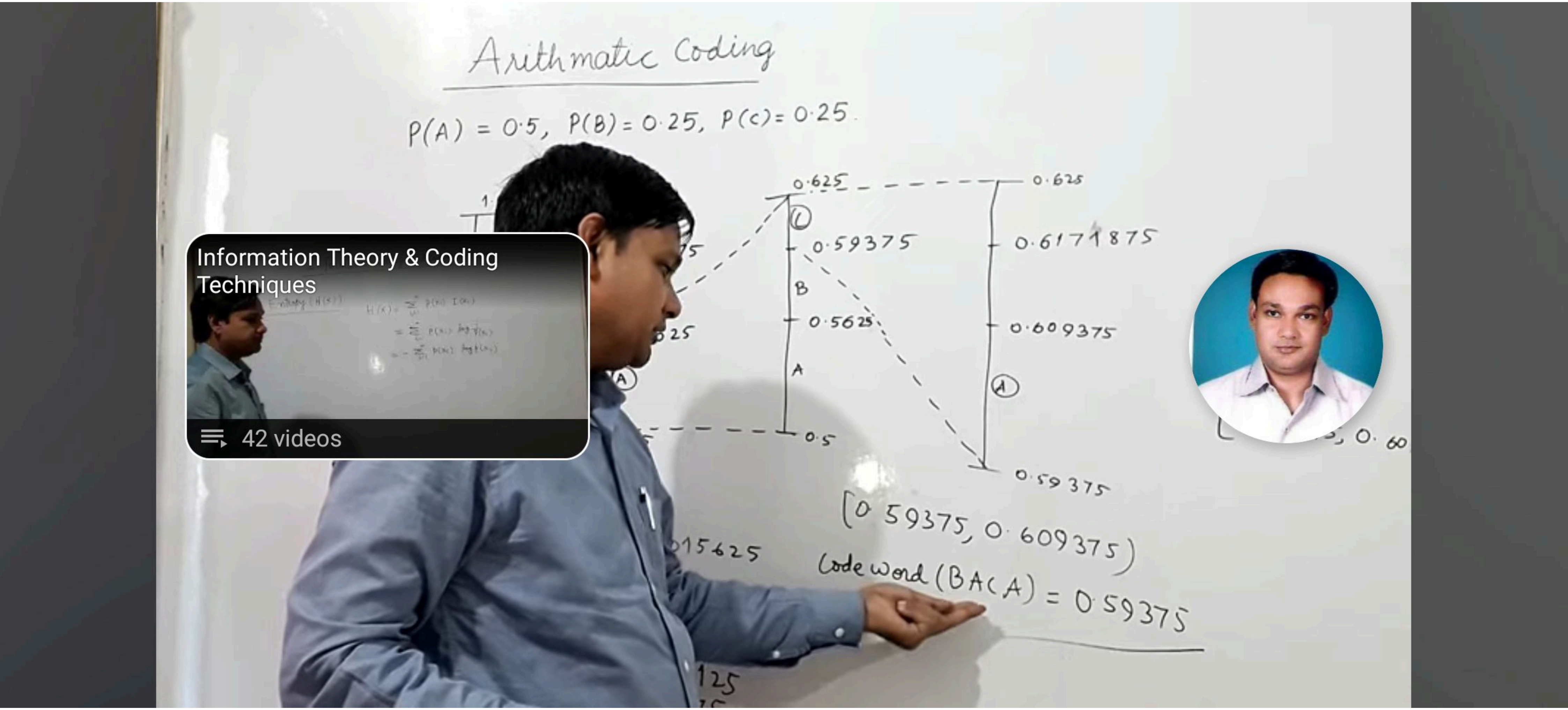
$$\rightarrow L = \sum p_i n_i$$
$$= 2(0.11 \times 1) + 0.11 \times 2 \times 7$$
$$= 1.76 \text{ bits/symbol}$$

$$\rightarrow \eta = \frac{H}{L \log_2 r} = \frac{3.15}{1.76 \log_2 4} = 0.8948$$

$$\sigma^2 = \sum p_i (n_i - L)^2$$
$$=$$

Arithmetic Coding


Arithmatic Coding
P(A) = 0.5, P(B) = 0.25, P(C) = 0.25
BACA
1.0
C
0.75
B
0.5
A
0
0.75
0.6875
B
0.625
A
0.5
0.625
C
0.59375
B
0.5625
A
0.5
0.625
0.6171875
0.609375
A
0.59375
×1/2 = 0.015625
+0.59375
0.609375
1/4 = 0.0078125
0.609375

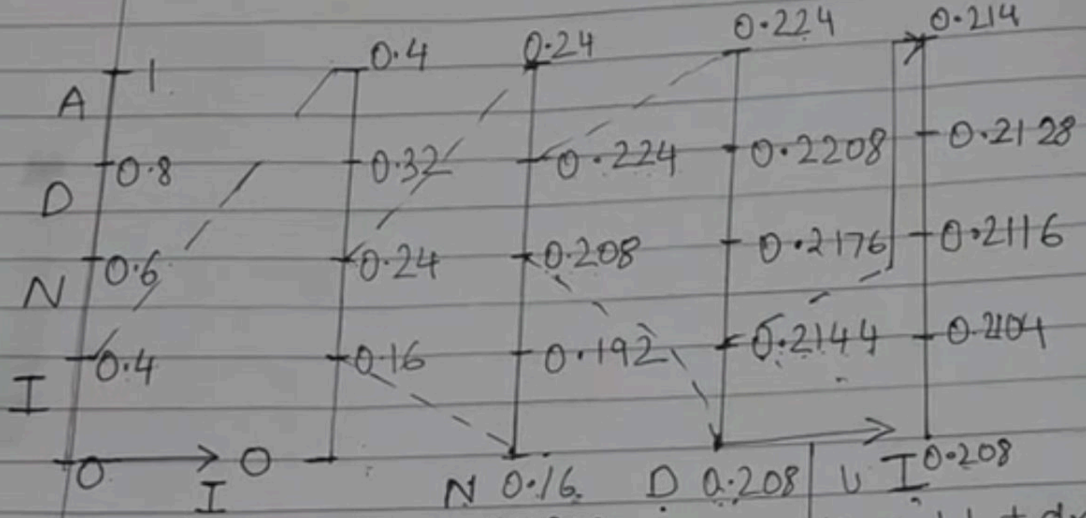

Arithmatic Coding
P(A) = 0.5, P(B) = 0.25, P(C) = 0.25
0.625
C
0.59375
B
0.5625
A
0.5
0.625
0.6171875
0.609375
A
0.59375
[0.59375, 0.609375)
Code word (BACA) = 0.59375
Information Theory & Coding Techniques
42 videos

ARITHMETIC CODING

message INDIA

Symbol	Prob	CDF
I	$2/5 = 0.4$	0.4
N	$1/5 = 0.2$	0.6
D	$1/5 = 0.2$	0.8
A	$1/5 = 0.2$	1

D	$1/5 = 0.2$	0.8
A	$1/5 = 0.2$	1



$UL = LL + d(P)(d)$

$= 0 + 0.4 \times 0.4$

$UL = LL + D \times P(d)$

$0 + 0.4 \times 0.6$

N $UL = LL + D \times P(d)$

$UL = LL + d \times P(d)$

$= 0.208 + 0.016 \times 0.4$

$UL = LL + d \times P(d)$

$= .208 + 0.006 \times 0.4$