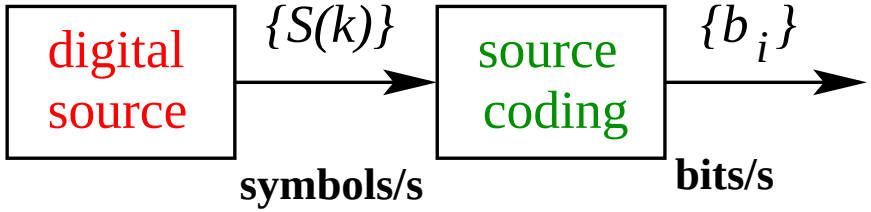
Information Theory Revision (Source)



- Digital source is defined by
  1. Symbol set: $\mathcal{S} = \{m_i, 1 \leq i \leq q\}$
  2. Probability of occurring of m_i: p_i, $1 \leq i \leq q$
  3. Symbol rate: R_s [symbols/s]
  4. Interdependency of $\{S(k)\}$
- Information content of alphabet m_i: $I(m_i) = -\log_2(p_i)$ [bits]
- Entropy: quantifies average information conveyed per symbol
  - Memoryless sources: $H = -\sum_{i=1}^{q} p_i \cdot \log_2(p_i)$ [bits/symbol]
  - 1st-order memory (1st-order Markov) sources with transition probabilities p_{ij}

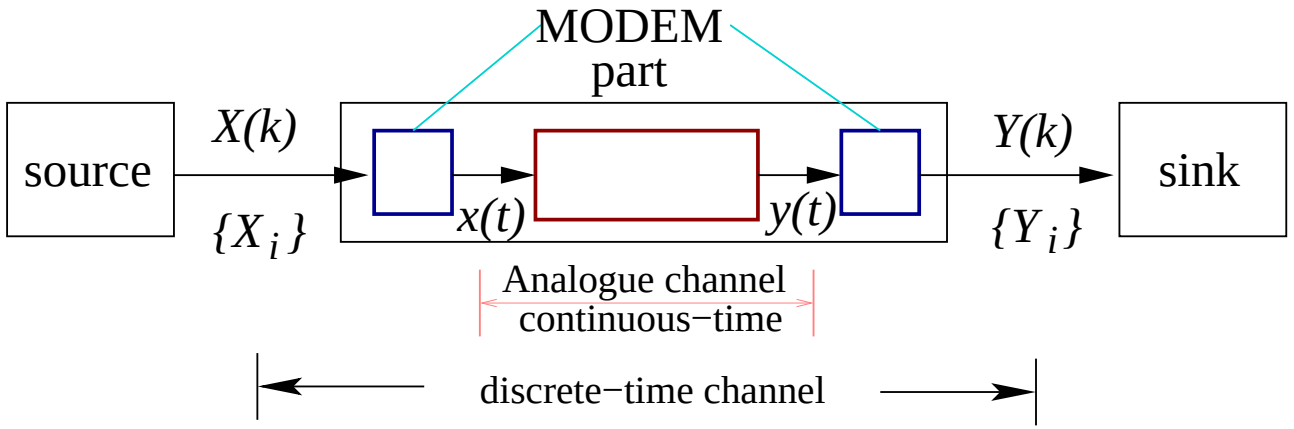
$$H = \sum_{i=1}^{q} p_i H_i = -\sum_{i=1}^{q} p_i \sum_{j=1}^{q} p_{ij} \cdot \log_2(p_{ij}) \text{ [bits/symbol]}$$

- Information rate: tells you how many bits/s information the source really needs to send out
  - Information rate $R = R_s \cdot H$ [bits/s]
- Efficient source coding: get rate R_b as close as possible to information rate R
  - Memoryless source: apply entropy coding, such as Shannon-Fano and Huffman, and RLC if source is binary with most zeros
  - Generic sources with memory: remove redundancy first, then apply entropy coding to "residauls"

Practical Source Coding

- Practical source coding is guided by information theory, with practical constraints, such as performance and processing complexity/delay trade off
- When you come to practical source coding part, **you can smile** – as you should know everything
- As we will learn, data rate is directly linked to required bandwidth, source coding is to encode source with a data rate as small as possible, i.e. as close to information rate as possible
- For correlated source or signal with memory, to achieve this
  1. Critical to remove redundancy or correlation of source signal first
  2. After removing predictable part, resulting residual signal is nearly white, can then be coded with entropy coding, such as Shannon-Fano or Huffman, or RLC if sequence is binary with most zeros
- Speech signal has significant temporal correlation
  - When you come to speech codecs, you can always identify 1. and 2., and see how temporal redundancy is dealt with in different practical ways
- Video signal has even more correlations: intra-frame (spatial) correlation and inter-frame (temporal) correlation
  - When you come to video codecs, you can always identify 1. and 2., and see how both spatial and temporal redundancies are dealt with in different practical ways
  - e.g. two consecutive video frames differ very little (temporal correlation): send a whole frame as reference, then only send difference of two consecutive frames – which may be coded by RLC

Information Theory Revision (Channel)



- Information theory studies what happens to information transferring across channel, and provides guiding principles for designing of practical communication systems
- Average mutual information $I(X,Y)$ between channel input $X(k) \in \{X_i\}$ and channel output $Y(k) \in \{Y_i\}$ characterises how information is transferring across channel

$$I(X,Y) = \sum_i \sum_j P(X_i, Y_j) \cdot \log_2 \frac{P(X_i|Y_j)}{P(X_i)} \qquad \text{(bits/symbol)}$$

  - What happens to information transferred across channel

$$\underbrace{I(X,Y)}_{\text{av. conveyed information}} = \underbrace{H(X)}_{\text{source entropy}} - \underbrace{H(X|Y)}_{\text{av. information lost}}$$

$$\underbrace{I(X,Y)}_{\text{av. conveyed information}} = \underbrace{H(Y)}_{\text{destination entropy}} - \underbrace{H(Y|X)}_{\text{error entropy}}$$

Channel Capacity

- Channel capacity C is maximum possible error-free information transmission rate across channel
- Channel capacity of discrete channel, where $T_{\rm av}$ is average symbol duration

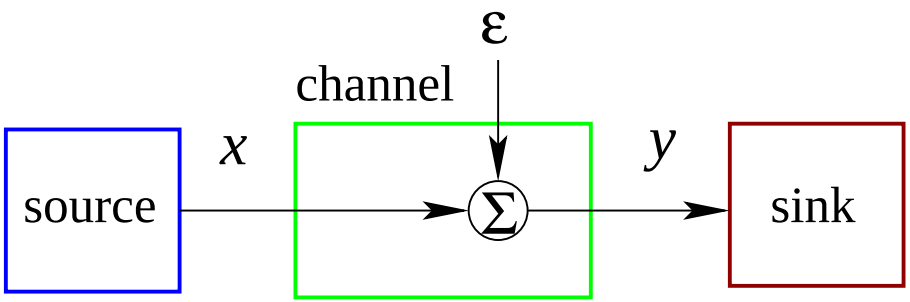
$$C = \max I(X, Y) \text{ [bits/symbol]} \quad \text{or } C = \max I(X, Y)/T_{\rm av} \text{ [bits/s]}$$

  - Channel capacity of BSC with error probability p_e

$$C = 1 + (1 - p_e)\log_2(1 - p_e) + p_e \log_2 p_e \text{ [bits/symbol]}$$

    Best case: $p_e = 0$ and $C = 1$ [bits/symbol]; worst case: $p_e = 0.5$ and $C = 0$ [bits/symbol]

- Gaussian channel capacity and Shannon-Hartley expression

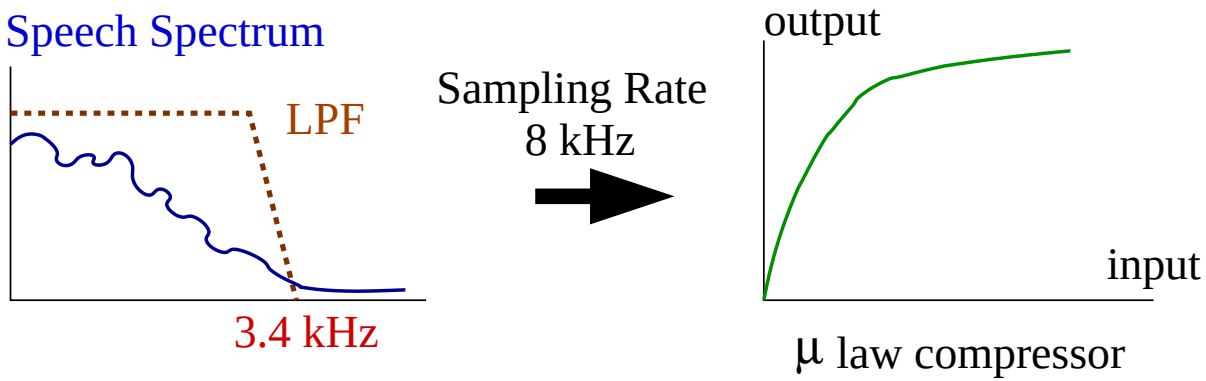


$$C = B \cdot \log_2\left(1 + \frac{S_P}{N_P}\right) \text{ [bits/s]}$$

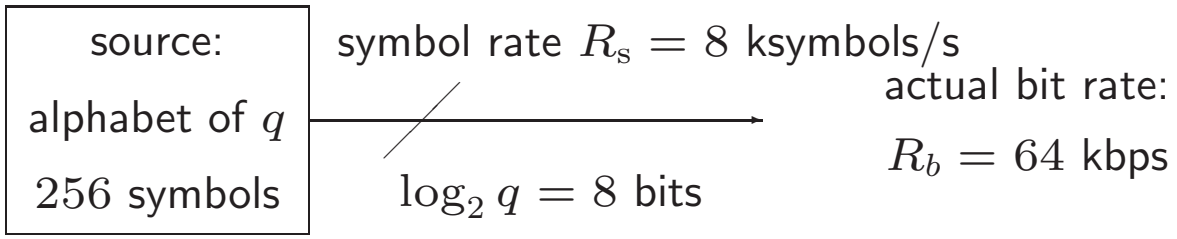
  - Two basic resources are channel bandwidth B [Hz] and signal power S_P
  - For AWGN ε with two-side power spectral density $N_0/2$, noise power $N_P = B \cdot N_0$
  - We can trade the channel bandwidth B for signal power S_P, and vice versa

60's 64kbps "Speech Codec"

- When analogue telephone network was digitised, the following 64 kbps "speech codec" was define
  - Analogue speech signal is passed through a low pass filter with cut-off frequency 3.4 KHz, and filtered signal is sampled at sampling rate 8 ksymbols/s
  - As speech signal samples are most have small magnitudes and large-magnitudes are rare, samples are passed through a μ-law compressor



  - Compressed samples are quantised by a 8-bit quantiser, i.e. quantised to 256 symbols (level)



- You have learnt information theory, what is your criticism of this "speech codec"

Critics

- Most of students simply work out this has a bit rate of 64 kbps – you was told this !
  - Most of you know low pass filtering is to limit speech signal bandwidth, and as most of speech energy is in low frequency band, cut off frequency of 3.4 kHz is appropriate
  - As filtered signal is band limited to 3.4 kHz, you know you must sample at least twice of this, so sampling rate of 8 kHz is appropriate
- Cleve ones then "spot" – speech samples (magnitudes) are not equiprobable, and BCD of 8 bits is not right, an entropy encoding should be used – this is actually not correct !
  - μ-law compressor compresses large-magnitude samples but it also expands small-magnitude samples, so it should be called compressor-expandor
  - compressed and expanded samples are more near equiprobable – you can see the μ-law compressor is some sort of practical implementation of entropy encoding
- So what is wrong? If speech source is memoryless, then all sounds great, or this "speech codec" treats speech as independent source, but speech samples are highly correlated !
  - First thing you should spot is the bit rate is far far bigger than really necessary, i.e. much much larger than information rate
  - Remove redundancy, e.g. build a predictive model, after remove predictable part, resulting residual sequence will be near white and can be coded with far far smaller bit rate
  - In practice, you of course need to send your predictor parameters, look at your mobile phone, its speech codec probably only has a few kbps bit rate

Example 1

1. A source emits symbols X_i, $1 \leq i \leq 6$, in the BCD format with probabilities $P(X_i)$ as given in Table 1, at a rate $R_{\rm s} = 9.6$ kbaud (baud=symbol/second).
   State (i) the information rate and (ii) the data rate of the source.

Table 1.

X_i	$P(X_i)$	BCD word
A	0.30	000
B	0.10	001
C	0.02	010
D	0.15	011
E	0.40	100
F	0.03	101

2. Apply Shannon-Fano coding to the source characterised in Table 1. Are there any disadvantages in the resulting code words?
3. What is the original symbol sequence of the Shannon-Fano coded signal 110011110000110101100?
4. What is the data rate of the signal after Shannon-Fano coding? What compression factor has been achieved?
5. Derive the coding efficiency of both uncoded BCD signal as well as Shannon-Fano coded signal.
6. Repeat parts 2 to 5 but this time with Huffman coding.

Keys to remember:

- Entropy coding is efficient for memoryless sources
- Assign number of bits to a symbol according to its information content
- Practical constraints: (a) practical coding cannot have a fraction of bit, i.e. number of bits assigned to a symbol must be an integer; (b) no codword forms a prefix for any other codeword

Example 1 - Solution

1. (i) Entropy of source:

$$\begin{aligned} H &= -\sum_{i=1}^{6} P(X_i) \cdot \log_2 P(X_i) = -0.30 \cdot \log_2 0.30 - 0.10 \cdot \log_2 0.10 - 0.02 \cdot \log_2 0.02 \\ &\quad -0.15 \cdot \log_2 0.15 - 0.40 \cdot \log_2 0.40 - 0.03 \cdot \log_2 0.03 \\ &= 0.52109 + 0.33219 + 0.11288 + 0.41054 + 0.52877 + 0.15177 \\ &= 2.05724 \text{ bits/symbol} \end{aligned}$$

Information rate: $R = H \cdot R_{\rm s} = 2.05724$ [bits/symbol] $\cdot$ 9600 [symbols/s] $= 19750$ [bits/s]
(ii) Data rate $= 3$ [bits/symbol] $\cdot\, 9600$ [symbols/s] $= 28800$ [bits/s]

2. Shannon-Fano coding:

X	$P(X)$	I (bits)	steps					code
E	0.4	1.32	0					0
A	0.3	1.74	1	0				10
D	0.15	2.74	1	1	0			110
B	0.1	3.32	1	1	1	0		1110
F	0.03	5.06	1	1	1	1	0	11110
C	0.02	5.64	1	1	1	1	1	11111

Disadvantage: the rare code words have maximum possible length of $q - 1 = 6 - 1 = 5$, and a buffer of 5 bits is required, compared with 3 bits for BCD

3. Shannon-Fano encoded sequence: $110|0|11110|0|0|0|110|10|110|0$ = DEFEEEDADE
4. Average code word length:

$$d = 0.4 \cdot 1 + 0.3 \cdot 2 + 0.15 \cdot 3 + 0.1 \cdot 4 + 0.05 \cdot 5 = 2.1 \quad \text{[bits/symbol]}$$

As a comparison, every code word in BCD has bits, or average code word length of 3 bits

Data rate of Shannon-Fano encoded signal:

$$d \cdot R_{\rm s} = 2.1 \cdot 9600 = 20160 \quad \text{[bits/s]}$$

Compression factor:

$$\frac{3 \ \text{[bits]}}{d \ \text{[bits]}} = \frac{3}{2.1} = 1.4286$$

5. Coding efficiency before Shannon-Fano:

$$\text{CE} = \frac{\text{information rate}}{\text{data rate}} = \frac{19750}{28800} = 68.58\%$$

Coding efficiency after Shannon-Fano:

$$\text{CE} = \frac{\text{information rate}}{\text{data rate}} == \frac{19750}{20160} = 97.97\%$$

Hence Shannon-Fano coding brought the coding efficiency close to 100%.

6. Huffman coding:

X	$P(X)$	steps 1	2	3	4	5	code
E	0.4					1	1
A	0.3				0	0	00
D	0.15			0	1	0	010
B	0.1		0	1	1	0	0110
F	0.03	0	1	1	1	0	01110
C	0.02	1	1	1	1	0	01111

step 1			step 2		
E	0.40		E	0.40	
A	0.30		A	0.30	
D	0.15		D	0.15	
B	0.10		B	0.10	0
F	0.03	0	FC	0.05	1
C	0.02	1			

step 3			step 4			step 5		
E	0.40		E	0.40		ADBFC	0.60	0
A	0.30		A	0.30	0	E	0.40	1
D	0.15	0	DBFC	0.30	1			
BFC	0.15	1						

Note reverse bit order in extracting code words

Same disadvantage as Shannon-Fano: the rare code words have maximum possible length of $q - 1 = 6 - 1 = 5$, and a buffer of 5 bit is required.

$1|1|00|1|1|1|1|00|00|1|1|010|1|1|00 =$ EEAEEEEAAEEDEEA

The same data rate and the same compression factor achieved as Shannon-Fano coding.

The coding efficiency of the Huffman coding is identical to that of Shannon-Fano coding.

Example 1 – A Few Comments

- Assign bit length according to information content – so this is more efficient than slide 93

X	$P(X)$	I (bits)	$\approx$ (bits)	steps				code
E	0.4	1.32	1	0				0
A	0.3	1.74	2	1	0			10
D	0.15	2.74	3	1	1	0		110
B	0.1	3.32	3	1	1	1		111
F	0.03	5.06	5	1	1	1	0	1110
C	0.02	5.64	5	1	1	1	1	1111

  - Incidentally, you can work out average code word length $d = 1.95$ bits/symbol, which is actually smaller than source entropy $H = 2.05724$ bits/symbol – so this is definitely wrong! – spot it?

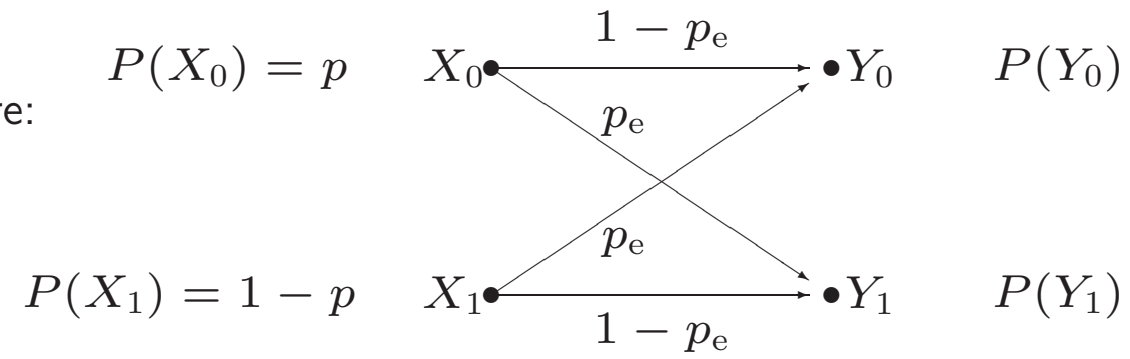
- Assign bit length according to information content – so this is more efficient than slide 93

X	$P(X)$	I (bits)	$\approx$ (bits)	steps				code
E	0.4	1.32	1	0				0
A	0.3	1.74	2	1	0			10
D	0.15	2.74	3	1	1	0	0	1100
B	0.1	3.32	3	1	1	0	1	1101
F	0.03	5.06	5	1	1	1	0	1110
C	0.02	5.64	5	1	1	1	1	1111

  - Compare with slide 93, D coded with one more bit but both F and C coded with one less bit
  - A good bargain? Not so! $P(D) = 0.15 \gg P(F) + P(C) = 0.05$, this is less efficient
  - In fact, average code word length is $d = 2.2$ bits/symbol, but slide 93 average code word length is $d = 2.1$ bits/symbol

Example 2

1. Considering the binary symmetric channel (BSC) shown in the figure:



   From the definition of mutual information,

$$I(X,Y) = \sum_i \sum_j P(X_i, Y_j) \cdot \log_2 \frac{P(X_i|Y_j)}{P(X_i)} \quad \text{[bits/symbol]}$$

   derive both

   (i) a formula relating $I(X,Y)$, the source entropy $H(X)$, and the average information lost per symbol $H(X|Y)$, and

   (ii) a formula relating $I(X,Y)$, the destination entropy $H(Y)$, and the error entropy $H(Y|X)$.

2. State and justify the relation ($>$,$<$,$=$,$\leq$, or $\geq$) between $H(X|Y)$ and $H(Y|X)$.

3. Considering the BSC in Figure 1, we now have $p = \frac{1}{4}$ and a channel error probability $p_e = \frac{1}{10}$. Calculate all probabilities $P(X_i, Y_j)$ and $P(X_i|Y_j)$, and derive the numerical value for the mutual information $I(X,Y)$.

Example 2 - Solution

1. (i) Relating to source entropy and average information lost:

$$
\begin{aligned}
I(X,Y) &= \sum_i \sum_j P(X_i,Y_j) \cdot \log_2 \frac{P(X_i|Y_j)}{P(X_i)} \\
&= \sum_i \sum_j P(X_i,Y_j) \cdot \log_2 \frac{1}{P(X_i)} - \sum_i \sum_j P(X_i,Y_j) \cdot \log_2 \frac{1}{P(X_i|Y_j)} \\
&= \sum_i \left(\sum_j P(X_i,Y_j) \right) \cdot \log_2 \frac{1}{P(X_i)} \\
&\quad - \sum_j P(Y_j) \cdot \left(\sum_i P(X_i|Y_j) \cdot \log_2 \frac{1}{P(X_i|Y_j)} \right) \\
&= \sum_i P(X_i) \cdot \log_2 \frac{1}{P(X_i)} - \sum_j P(Y_j) \cdot I(X|Y_j) = H(X) - H(X|Y)
\end{aligned}
$$

(ii) Bayes rule :

$$
\frac{P(X_i|Y_j)}{P(X_i)} = \frac{P(X_i,Y_j)}{P(X_i) \cdot P(Y_j)} = \frac{P(Y_j|X_i)}{P(Y_j)}
$$

Hence, relating to destination entropy and error entropy:

$$I(X,Y) = \sum_i \sum_j P(X_i,Y_j) \cdot \log_2 \frac{P(Y_j|X_i)}{P(Y_j)} = \sum_i \sum_j P(Y_j,X_i) \cdot \log_2 \frac{1}{P(Y_j)}$$

$$- \sum_i \sum_j P(Y_j,X_i) \cdot \log_2 \frac{1}{P(Y_j|X_i)} = H(Y) - H(Y|X)$$

2. Unless $p_e = 0.5$ or for equiprobable source symbols X, the symbols Y at the destination are more balanced, hence $H(Y) \geq H(X)$. Therefore, $H(Y|X) \geq H(X|Y)$.
3. Joint probabilities:

$$P(X_0,Y_0) = P(X_0) \cdot P(Y_0|X_0) = \frac{1}{4} \cdot \frac{9}{10} = 0.225$$

$$P(X_0,Y_1) = P(X_0) \cdot P(Y_1|X_0) = \frac{1}{4} \cdot \frac{1}{10} = 0.025$$

$$P(X_1,Y_0) = P(X_1) \cdot P(Y_0|X_1) = \frac{3}{4} \cdot \frac{1}{10} = 0.075$$

$$P(X_1,Y_1) = P(X_1) \cdot P(Y_1|X_1) = \frac{3}{4} \cdot \frac{9}{10} = 0.675$$

Destination total probabilities:

$$P(Y_0) = P(X_0) \cdot P(Y_0|X_0) + P(X_1) \cdot P(Y_0|X_1) = \frac{1}{4} \cdot \frac{9}{10} + \frac{3}{4} \cdot \frac{1}{10} = 0.3$$

$$P(Y_1) = P(X_0) \cdot P(Y_1|X_0) + P(X_1) \cdot P(Y_1|X_1) = \frac{1}{4} \cdot \frac{1}{10} + \frac{3}{4} \cdot \frac{9}{10} = 0.7$$

Conditional probabilities:

$$P(X_0|Y_0) = \frac{P(X_0,Y_0)}{P(Y_0)} = \frac{0.225}{0.3} = 0.75$$

$$P(X_0|Y_1) = \frac{P(X_0,Y_1)}{P(Y_1)} = \frac{0.025}{0.7} = 0.035714$$

$$P(X_1|Y_0) = \frac{P(X_1,Y_0)}{P(Y_0)} = \frac{0.075}{0.3} = 0.25$$

$$P(X_1|Y_1) = \frac{P(X_1,Y_1)}{P(Y_1)} = \frac{0.675}{0.7} = 0.964286$$

Mutual information:

$$I(X,Y) = P(X_0,Y_0)\cdot\log_2\frac{P(Y_0|X_0)}{P(Y_0)} + P(X_0,Y_1)\cdot\log_2\frac{P(Y_1|X_0)}{P(Y_1)}$$

$$+ P(X_1,Y_0)\cdot\log_2\frac{P(Y_0|X_1)}{P(Y_0)} + P(X_1,Y_1)\cdot\log_2\frac{P(Y_1|X_1)}{P(Y_1)}$$

$$= 0.3566165 - 0.0701838 - 0.1188721 + 0.2447348 = 0.4122954 \text{ [bits/symbol]}$$

- Key point regarding average mutual information

$$\underbrace{I(X,Y)}_{\text{av. conveyed information}} = \underbrace{H(X)}_{\text{source entropy}} - \underbrace{H(X|Y)}_{\text{av. information lost}}$$

Example 3

- A digital communication system uses a 4-ary signalling scheme with symbol set $\{X_1 = -\mathbf{3}, X_2 = -\mathbf{1}, X_3 = \mathbf{1}, X_4 = \mathbf{3}\}$. The channel is an ideal additive white Gaussian noise (AWGN) channel, the transmission rate is 100 MBaud (10^8 symbols/s) and the channel signal to noise ratio is known to be 63. The probabilities of occurrence for the four symbols at transmitter are respectively

$$P(X_1) = 0.2,\ P(X_2) = 0.3,\ P(X_3) = 0.2,\ P(X_4) = 0.3$$

It is known that this 4-ary symbol source is a first-order Markov process with the known transition probability matrix

$$\mathbf{\Gamma} = P\left[p_{i,j}\right] = \begin{bmatrix} 0.6 & 0.1 & 0.1 & 0.2 \\ 0.1 & 0.6 & 0.1 & 0.2 \\ 0.1 & 0.1 & 0.8 & 0.0 \\ 0.0 & 0.1 & 0.1 & 0.8 \end{bmatrix}$$

where $p_{i,j} = P(X_j|X_i)$, $1 \leq i, j \leq 4$

1. Determine the information rate of this 4-ary source.
2. If you are able to employ some capacity-approaching error-correction coding technique and would like to achieve error-free transmission, what is the minimum channel bandwidth required?

Error-free transmission only possible if source information rate does not exceed channel capacity C

$$C = B \cdot \log_2(1 + \text{SNR})$$

Example 3 - Solution

1. Symbol entropy for X_i, $1 \leq i \leq 4$

$$\begin{aligned} H_1 = H_2 &= -0.2 \cdot \log_2 0.2 - 2 \cdot 0.1 \cdot \log_2 0.1 - 0.6 \cdot \log_2 0.6 \\ &= 0.4643856 + 0.6643856 + 0.4421793 = 1.5709505 \quad \text{[bits/symbol]} \\ H_3 = H_4 &= -2 \cdot 0.1 \cdot \log_2 0.1 - 0.8 \cdot \log_2 0.8 \\ &= 0.6643856 + 0.2575424 = 0.921928 \quad \text{[bits/symbol]} \end{aligned}$$

Source entropy:

$$H = \sum_{i=1}^{4} P_i \cdot H_i = 0.5 \cdot 1.5709505 + 0.5 \cdot 0.921928 = 1.2464393 \quad \text{[bits/symbol]}$$

Source information rate:

$$R = H \cdot R_{\rm s} = 1.2464393 \times 10^8 = 124.6439 \quad \text{[Mbits/s]}$$

2. To achieve error-free transmission:

$$R \leq C = B \log_2 \left(1 + \frac{S_P}{N_P}\right) \longrightarrow 124.6439 \times 10^6 \leq B \log_2(1 + 63)$$

Thus

$$B \geq 21 \quad \text{[MHz]} \longrightarrow B_{\rm min} = 21 \quad \text{[MHz]}$$

Example 4

- A predictive source encoder generates a bit stream at a bit rate of 3.1844 Mbits/s, and it is known that the probability of a bit taking the value **0** is $P(\mathbf{0}) = p = 0.95$. The bit stream is then encoded by a run length encoder (RLC) with a codeword length of $n = 5$ bits.

1. Determine the compression ratio of the RLC, and the bit rate after the RLC.
2. Find the encoder input patterns that produce the following encoder output cordwords

$$11111 \quad 11110 \quad 11101 \quad 11100 \quad 11011 \quad \cdots \quad 00001 \quad 00000$$

3. What is the encoder input sequence of the RLC coded signal 110110000011110?

- Key points regarding run length encoder
  - RLC is beneficial for binary sources with probability of bit **0** much larger than that of bit **1**, i.e. binary sequences with most zeros
  - Compression ratio is the ratio of the total number of bits before the encoder to the total number of bits after the encoder, which is equal to

$$\text{Compression ratio} = \frac{d}{n}$$

where d is the average codeword length before the encoder and n is the average codeword length after the encoder

Example 4 - Solution

1. Codeword length after RLC is $n = 5$ bits, and average codeword length d before RLC with $N = 2^n - 1$

$$d = \sum_{l=0}^{N-1} (l+1) \cdot p^l \cdot (1-p) + N \cdot p^N = \frac{1-p^N}{1-p}$$

Compression ratio

$$\frac{d}{n} = \frac{1-p^N}{n(1-p)} = \frac{1-0.95^{31}}{5 \times 0.05} = 3.1844$$

The bit rate after the RLC

$$R_{\text{RLC}} = \frac{3.1844 \text{ [Mbits/s]}}{\text{Compression ratio}} = 1 \text{ [Mbits/s]}$$

2. RLC table

$$\underbrace{00\cdots00}_{31}00000 \rightarrow 11111 \qquad \underbrace{00\cdots00}_{30}0000\,1 \rightarrow 11110$$

$$\underbrace{00\cdots00}_{29}000\,1 \rightarrow 11101 \qquad \underbrace{00\cdots00}_{28}00\,1 \rightarrow 11100$$

$$\underbrace{00\cdots00}_{27}0\,1 \rightarrow 11011 \qquad \cdots$$

$$01 \rightarrow 00001 \qquad 1 \rightarrow 00000$$

3. $11011 \mid 00000 \mid 11110 \longleftarrow$ the encoder input sequence

$$\underbrace{00\cdots00}_{27}0\,1\;1\;\underbrace{00\cdots00}_{30}0000\,1$$