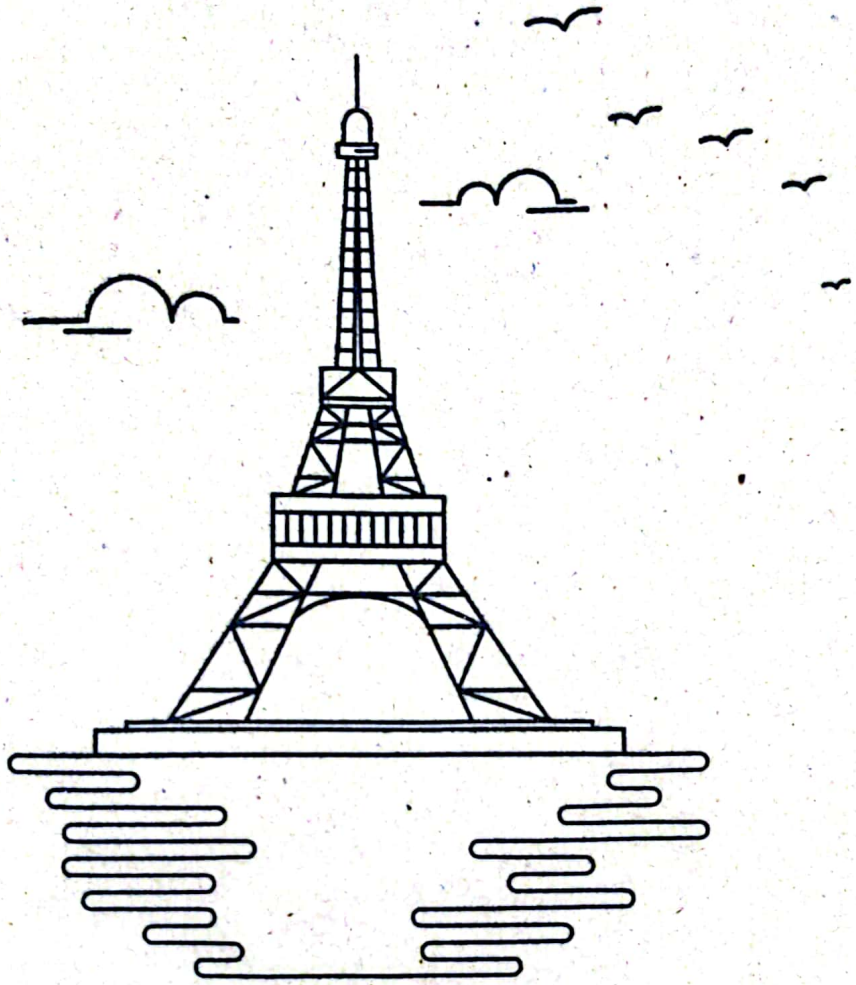
CLASSMATE



Eiffel Tower, France

EXERCISE BOOK

Information Theory and Coding

ETE 311

CamScanner

24 Oct.

Reference Book

Information theory, Coding and Cryptography - R. Bose

Information Theory: Information theory is a mathematical approach to the study of coding of information, along with the quantification storage and communication of information.

Limitation of communication:

i) **Limited Bandwidth:** The information theory states that the greater is the transmission bandwidth of a communication system, the more is the information that can be transmitted. But there is always a limitation in bandwidth.

The complete amount of information available to the human ear is contained the frequency range up to 15KHz. However, in AM radio the maximum modulating frequency is restricted up to 5KHz, hence the bandwidth is 10KHz. Therefore, AM radio receiver cannot reproduce all the information contained in the music because this will require a bandwidth of 30KHz.

ii) **Inevitable Background Noise:** In long distance communication systems, operating with limited amount of signal power, the signal may be as small as noise or even smaller than the noise. Thus, in such cases, the presense of noise severals limits the capabilities of a communication system.

iii) **Limited Power:** With the increase of power, the strength of signal will also be increased. But it is difficult to increase power unlimitedly as increased power causes increase

of noise and also increasing power is difficult as it is very costly. It is a limitation of the communication system.

"If the rate of information from a source doesn't exceed the capacity of the channel, then there exist a coding system such that information can be transmitted over the communication channel with arbitary small amount of errors despite the presense of noise."

Information Measure:

$$\text{Information} \propto \frac{1}{\text{Probability of occurance}}$$

Hence, probability high → information less
probability low → information maximum

Self-Information:

Self-information is a measure of the information content associated with the outcome of a random variable.

Self information can be calculated as a discrete event x as follows:

$$I(x_i) = -\log_2 P(x_i)$$

If two discrete event $P(x_1) = 0.25$ and $P(x_2) = 0.5$, Calculate the self information.

→ Self information, $I(x_i) = -\log_2 P(x_1) - \log_2 P(x_2)$

$= 2 + 1$

$= 3$

Properties of self-information:

$I(x_i) = 0$, If $P_i = 1$

$I(x_i) \geq 0$, For $0 \leq P(x_i) \leq 1$ [Self information Negative না]

$I(x_1) > I(x_2)$, if $P_1 < P_2$ [Inverse Proportional]

$I(x_1 \cdot x_2) = I(x_1) + I(x_2)$

Entropy:

Entropy is the measurement of uncertainity of any event can be defined as

Element → $H(x_i) = -P_i \log_2 P_i$ where $x_i \in X$ & $\sum P_i = 1$

Full → $H(X) = \sum_{x_i \in X} H(x_i) = -\sum_{i=1}^{n} P_i \log_2 P_i$ bit/symbol

The entropy of a random variable x with a probability mass function $P(x)$ is -

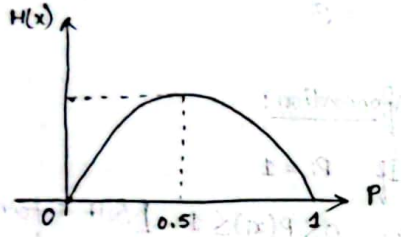
$$H(x) = -\sum P(x) \cdot \log_2 P(x)$$

CamScanner

Properties of Entropy:

i) Entropy is non-negative

ii) Entropy is maximum for equi probable events



i) Entropy is concave function

ii) When $P = 0, 1$; there is no uncertainity, Entropy $= 0$

iii) When $P = 0.5$; Entropy is maximum

Uncertainity is maximum

$$H(x) = \sum_{k=1}^{M} P_k \log_2 \frac{1}{P_k}$$

$$= M \cdot \frac{1}{M} \log_2 \frac{1}{\frac{1}{M}}$$

$$= \log_2 M$$

If the QPSK modulated signal is

00 01 10 10 11 11 01 01 10 00 10 10 11 01 10

$P_{00} = \frac{2}{15}$ $P_{01} = \frac{4}{15}$ $P_{10} = \frac{6}{15}$ $P_{11} = \frac{3}{15}$

So, The entropy is

$$H(x) = -\sum P(x) \cdot \log P(x)$$

$$= -P_{00} \log_2 P_{00} - P_{01} \log_2 P_{01} - P_{10} \log_2 P_{10} - P_{11} \log_2 P_{11}$$

$$= -\frac{2}{15} \log_2 \frac{2}{15} - \frac{4}{15} \log \frac{4}{15} - \frac{6}{15} \log_2 \frac{6}{15} - \frac{3}{15} \log_2 \frac{3}{15}$$

$$= 1.889 \text{ unit}$$

For BPSK Signal

$$x = \begin{cases} 1 & ; \text{Probability } P \\ 0 & ; \text{Probability } 1-P \end{cases}$$

So, $H(x) = -\sum_{x=0}^{1} P(x) \log_2 P(x)$

or, $H(P) = -P \log_2 P - (1-P) \log_2 (1-P)$

When $P = \frac{1}{2}$, $H(x) = 1$

For, $P = 0.5$, Entropy is maximum ; $H(x) = 1$

$P = 0, 1$; There is no uncertainity

Entropy is zero if one outcome certain, Entropy is maximum if both outcomes are equally likely.

CamScanner

Joint Entropy

Joint entropy $H(x,y)$ of a pair of discrete random variable (x,y) with joint distribution $P(x,y)$ is defined as -

$$H(x,y) = -\sum_{x\in X}\sum_{y\in Y} P(x,y) \cdot \log P(x,y)$$

or, $H(x,y) = -\sum \log P(x,y)$

Conditional Entropy

The conditional entropy of a random variable given another as the expected value of the entropies of the conditional distribution averaged over conditioning random variable.

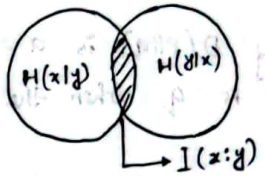
If $H(x|y) \sim P(x,y)$, The conditional entropy $H(y|x)$ is defined as -

$$H(y|x) = \sum_{x\in X} P(x) \cdot H(y|x=x)$$

$$= -\sum_{x\in X} P(x) \cdot \sum_{y\in Y} P(y|x) \cdot \log P(y|x)$$

$$= -\sum_{x\in X}\sum_{y\in Y} P(x,y) \cdot \log P(y|x)$$

$$= -E_{P(x,y)} \cdot \log P(y|x)$$ [Bayes Rule অনুযায়ী]



Physical Significance of Mutual Information

- If x and y are independent, then their joint entropy is the sum of individual entropy

$$H(x,y) = H(x) + H(Y)$$

- If x and y are dependent, the degree of their ~~mutual~~ dependence can be measured by mutual information

$$I(x:y) = H(x) + H(y) - H(x,y)$$

- If $I(x:y)$ measures the correlation between x and y
  - Equals to zero if and only if x and y are independent
- If $(x:y)$ measures the uncertainity reduction in x, if we know y

$$I(x:y) = H(y) - H(y|x)$$
$$= H(x) - H(x|y)$$

CamScanner

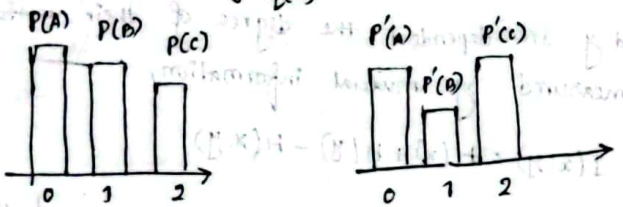
Relative Entropy:

The relative entropy $D(p\|q)$ is a measure of assuming that the distribution is q when the true distribution is p.

The relative entropy is the distance between two probability mass function $P(x)$ and $q(x)$ which is defined as -

$$D(p\|q) = \sum_{x \in X} P(x) \log \frac{P(x)}{q(x)}$$

$$= E_p \log \frac{P(x)}{q(x)}$$



Significance of Relative Entropy

Relative entropy is always non-negative and is zero if $p=q$. However it is not true distance between distributions since it is not symmetric and doesn't satisfy the triangle inequality. Nonetheless, it is often useful to think of relative entropy as a distance between distribution.

Mutual Information:

Mutual information is a measure of the amount of information that one random variable contains about another random variable.

The mutual information $I(x:y)$ is the relative entropy between the joint distribution and product distribution $P(x)$, $P(y)$

$$I(x:y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) \log \frac{P(x,y)}{P(x) \cdot P(y)}$$

$$= D(P(x,y) \| P(x) \cdot P(y))$$

$$= E_{P(x,y)} \cdot \log \frac{P(x,y)}{P(x) \cdot P(y)}$$

Prove that, $D(p\|q) \neq D(q\|p)$

$\Rightarrow$ $X = \{0, 1\}$

Let, $P(0) = 1-r$

$P(1) = r$

$q(0) = 1-s$

$q(1) = s$

$$\therefore D(p\|q) = (1-r) \log \frac{1-r}{1-s} + r \cdot \log \frac{r}{s}$$

$$D(q\|p) = (1-s) \log \frac{1-s}{1-r} + s \cdot \log \frac{s}{r}$$

CamScanner

If $r = s$.

then $D(p\|q) = D(q\|p) = 0$

If $r = \frac{1}{2}$, $s = \frac{1}{4}$,

$$D(p\|q) = \frac{1}{2}\log\frac{1/2}{3/4} + \frac{1}{2}\log\frac{1/2}{1/4}$$

$$= \frac{1}{2}\log\left(\frac{2}{3}\right) + \frac{1}{2}\log(2)$$

$$= 0.2075 \text{ units}$$

$$D(q\|p) = \frac{3}{4}\log\frac{3/4}{1/2} + \frac{1}{4}\log\frac{1/4}{1/2}$$

$$= \frac{3}{4}\log\left(\frac{3}{2}\right) + \frac{1}{4}\log\left(\frac{1}{2}\right)$$

$$= 0.1887 \text{ units}$$

$\therefore D(p\|q) \neq D(q\|p)$

Chain rule theorem

The entropy of a pair of random variables is the entropy of one plus conditional entropy of the other.

$$H(x,y) = H(x) + H(y|x)$$

Proof:

$$H(x,y) = -\sum_{x\in X}\sum_{y\in Y} P(x,y)\log P(x,y)$$

$$= -\sum_{x\in X}\sum_{y\in Y} P(x,y)\cdot\log\left[P(x)\cdot P(y|x)\right]$$

$$= -\sum_{x\in X}\sum_{y\in Y} P(x,y)\cdot\log P(x) - \sum_{x\in X}\sum_{y\in Y} P(x,y)\log P(y|x)$$

$$= -\sum_{x\in X} P(x)\cdot\log P(x) - \sum_{x\in X}\sum_{y\in Y} P(x,y)\cdot\log P(y|x)$$

$$= H(x) + H(y|x)$$

$\therefore H(x,y) = H(x) + H(y|x)$

[Proved]

Bayes Rule:

i) $P(x|y) = \dfrac{P(x,y)}{P(y)}$

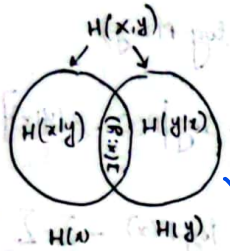
ii) $P(x,y) = P(x)\,P(y|x)$

$= P(y)\,P(x|y)$

$H(x,y) = H(x) + H(y|x) \rightarrow$ If x, y are independent, the entropy just adds

$\therefore H(x,y) = H(x) + H(y)$

Relationship between Entropy and Mutual Information:



$$I(x:y) = \sum P(x,y) \log \frac{P(x,y)}{P(x)\cdot P(y)}$$

$$= \sum_{x,y} P(x,y) \cdot \log \frac{P(x|y)}{P(x)}$$

$$= -\sum_{x,y} P(x,y) \cdot \log P(x) + \sum_{x,y} P(x,y) \cdot \log P(x|y)$$

$$= -\sum_{x} P(x) \cdot \log P(x) + \sum_{x,y} P(x,y) \log P(x|y)$$

$$= -\sum_{x} P(x) \cdot \log P(x) - \left(-\sum_{x,y} P(x,y) \log P(x|y)\right)$$

$$\therefore I(x:y) = H(x) - H(x|y) \quad \text{—(i)}$$

Thus, the mutual information $I(x:y)$ is the reduction in the uncertainity of x due to the knowledge of y.

Similarly, $I(x:y) = H(y) - H(x|y)$ —(ii)

We know,

$$H(x:y) = H(x) + H(y|x) \quad \text{—(iii)}$$

$$\therefore I(x:y) = H(y) - H(x,y) + H(x)$$

$$= H(x) + H(y) - H(x,y)$$

$\therefore$ Self information, $I(x,x) = H(x) - H(x|x)$

$$= H(x)$$

- Random variable is defined by previous known probability distribution

- Continuous Random variable – Sound frequency
  Discrete " " – ASCII Code

- **Information Rate:** Average number of bits of information per second

Consider a source is emitting $n \gg 1$ symbols

$\therefore$ The total number of information emitted is $nH(x)$

If the source generates r symbols per second the whole emitted sequence takes n/r seconds.

Therefore, the information generates at a rate of R

$$\frac{nH(x)}{n/r} = rH(x) \text{ bps}$$

$\frac{\text{Total information}}{\text{Total time}}$

$\therefore R = nH$ bps ; when n = generated message per second

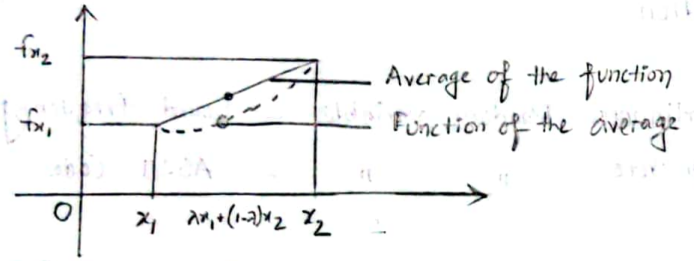
Properties of Convex function

i) 2nd Derivative Non-negative

ii) Tangent line [illegible]

iii) Average of the function > Function of the average

Given function $f(x) = x^2$



Average of the function:

$x = 0$, $f(x) = 0$

$x = 2$, $f(x) = 4$

$\therefore$ Average of the function $= \frac{0+4}{2} = 2$

Function of the average:

$x = \overset{a}{0}, \overset{b}{2}$

Average of $x = \frac{0+2}{2} = 1$

$\therefore$ function of the average $= f(1) = 1^2 = 1$

Jensen's Inequality:

i) A function $f(x)$ is said to be convex over an interval (a,b) if for every $x_1, x_2 \in (a,b)$ and $0 \le \lambda \le 1$, the following inequlity holds $f(\lambda x_1 + (1-\lambda)x_2) \le \lambda f(x_1) + (1-\lambda) f(x_2)$

or,

ii) A function $f(x)$ is said to be convex over ant interval (a,b) if for every $x_1, x_2 \in (a,b)$ and with probability P_1, P_2 where $P_1 + P_2 = 1$, $0 \le P_1 P_2 \le 1$, $F(P_1 x_1 + P_2 x_2) \le P_1 F(x_1) + P_2 F(x_2)$

According to information theory -

"If F is a convex function and X is a random variable then - $E\{f(x)\} \ge f(E\{x\})$"

(i) $E\{f(x)\} \ge f(E\{x\})$

(ii) $f(E\{x\}) \le E\{f(x)\}$

(iii) $\sum_{i=1}^{k} P_i f(x_i) \ge f\left(\sum_{i=1}^{k} P_i x_i\right)$

Physical Significance of Jensen's Inequility

i) Transmitted data of the average and average of the transmitted data can be calculated

ii) It is used to make claims about a function where little is known or needs to be known about the distribution.

CamScanner

Information inequility : $D(p\|q) \geq 0$

Proof: $D(p\|q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}$

$$\Rightarrow -D(p\|q) = -\sum_{x \in X} p(x) \cdot \log \frac{p(x)}{q(x)}$$

$$= \sum_{x \in X} p(x) \log \frac{q(x)}{p(x)}$$

$$\leq \log \sum_{x \in X} p(x) \frac{q(x)}{p(x)} \quad [E\{f(x)\} \leq f E\{x\} \text{ for concave function}]$$

$$\leq \log \sum_{x \in X} q(x)$$

$$\leq \log 1$$

$$\leq 0$$

Here, $\sum_{x \in X} q(x) = c \sum_{x \in X} p(x) = c$. We have inequality only if $\sum_{x \in X} q(x) = 1$ which implies that $c = 1$. Hence we have $D(p\|q) = 0$ if and only if $p(x) = q(x)$ for all x.

Q: How to reduce loss function using KL divergence or relative entropy?

A: We can reduce loss function by optimizing the parameters.

Suppose, $p(x) = 1$, $p'(x) = 0.1$

$p(y) = 0$, $p'(y) = 0.9$

$$\therefore D(p\|p') = p(x) \log \frac{p(x)}{p'(x)} + p(y) \log \frac{p(y)}{p'(y)}$$

$$= 1 \times \log \frac{1}{0.1} + 0$$

$$= 3.32 \text{ bit.}$$

But if we take $p'(x) = 0.9$ and $p'(y) = 0.1$,

$$\therefore D(p\|p') = p(x) \log \frac{p(x)}{p'(x)} + p(y) \log \frac{p(y)}{p'(y)}$$

$$= 1 \times \log \frac{1}{0.9} + 0$$

$$= 0.1$$

Where relative entropy tends to zero.

Let (X, Y) have the following distribution -

Y \ X	1	2	3	4
1	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{32}$
2	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{32}$	$\frac{1}{32}$
3	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$
4	$\frac{1}{4}$	0	0	0

Determine $H(X)$, $H(Y)$, $H(X,Y)$, $H(X|Y)$, $H(Y|X)$

Solution:

The marginal distribution of X is $P(x_1)=\frac{1}{2}$, $P(x_2)=\frac{1}{4}$, $P(x_3)=\frac{1}{8}$ and $P(x_4)=\frac{1}{8}$. and the marginal distribution of Y is $P(y_1)=1/4$, $P(y_2)=\frac{1}{4}$, $P(y_3)=\frac{1}{4}$ and $P(y_4)=\frac{1}{4}$.

$$\therefore H(x) = \sum_{i=1}^{4} P(x) \log_2 \frac{1}{P(x)}$$

$$= -\sum_{i=1}^{4} P(x) \log_2 P(x)$$

$$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{8}\log_2\frac{1}{8} - \frac{1}{8}\log_2\frac{1}{8}$$

$$= \frac{1}{2}+\frac{1}{2}+\frac{3}{8}+\frac{3}{8}$$

$$= \frac{14}{8} \text{ bit}$$

$$= \frac{7}{4} \text{ bit}$$

$$\therefore H(Y) = -\sum_{i=1}^{4} P(Y) \log_2 P(Y)$$

$$= -\frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} - \log_2\frac{1}{4}$$

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

$$= 2$$

$$\therefore H(X|Y) = \sum_{i=1}^{4} P(Y=i)\, H(X|Y=i)$$

$$= \sum P(Y=1) H(X|Y=1) + \sum P(Y=2) H(X|Y=2) + \sum P(Y=3) H(X|Y=3) + \sum P(Y=4) H(X|Y=4)$$

$$= \frac{1}{4} H\left(\frac{1}{2},\frac{1}{4},\frac{1}{8},\frac{1}{8}\right) + \frac{1}{4} H\left(\frac{1}{4},\frac{1}{2},\frac{1}{8},\frac{1}{8}\right) + \frac{1}{4} H\left(\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{1}{4}\right) + \frac{1}{4} H(1,0,0,0)$$

$$= \frac{1}{4}\times\left(\frac{1}{2}+\frac{1}{2}+\frac{3}{8}+\frac{3}{8}\right) + \frac{1}{4}\left(\frac{1}{2}+\frac{1}{2}+\frac{3}{8}+\frac{3}{8}\right) + \frac{1}{4}\times\left(\frac{1}{2}+\frac{1}{2}+\frac{1}{2}+\frac{1}{2}\right) + \frac{1}{4}\times(0+0+0+0)$$

$$= \frac{1}{4}\times\frac{14}{8} + \frac{1}{4}\times\frac{14}{8} + \frac{1}{4}\times 2 + 0 \text{ bits}$$

$$= \frac{11}{8} \text{ bits}$$

$$\frac{H(X,Y)}{H(Y)} = H(X|Y)$$

$$\frac{H(X,Y)}{H(X)} = H(Y|X)$$

$$\therefore H(Y|X) = \sum_{i=1}^{4} P(X=i)\, H(Y|X=i)$$

$$= \sum P(X=1) H(Y|X=1) + \sum P(X=2) H(Y|X=2) + \sum P(X=3)(Y|X=3) + \sum P(X=4)(Y|X=4)$$

$$= \frac{1}{2}\cdot H\left(\frac{1}{4},\frac{1}{8},\frac{1}{8},\frac{1}{2}\right) + \frac{1}{4} H\left(\frac{1}{4},\frac{1}{2},\frac{1}{4},0\right) + \frac{1}{8} H\left(\frac{1}{4},\frac{1}{4},\frac{1}{2},0\right) + \frac{1}{8} H\left(\frac{1}{4},\frac{1}{4},\frac{1}{2},0\right)$$

$$= \frac{1}{2}\left(\frac{1}{2}+\frac{3}{8}+\frac{3}{8}+\frac{1}{2}\right) + \frac{1}{4}\left(\frac{1}{2}+\frac{1}{2}+\frac{1}{2}+0\right) + \frac{1}{8}\left(\frac{1}{2}+\frac{1}{2}+\frac{1}{2}+0\right) + \frac{1}{8}\left(\frac{1}{2}+\frac{1}{2}+\frac{1}{2}+0\right)$$

CamScanner

$$= \frac{1}{2} \times \frac{14}{8} + \frac{1}{4} \times \frac{3}{2} + \frac{1}{8} \times \frac{3}{2} + \frac{1}{8} \times \frac{3}{2}$$

$$= \frac{13}{8}$$

$$\therefore H(X,Y) = H(X) + H(Y|X)$$

$$= \frac{7}{4} + \frac{13}{8}$$

$$= \frac{14+13}{8}$$

$$= \frac{27}{8} \text{ bits}$$

$$\text{or, } H(X,Y) = H(Y) + H(X|Y)$$

$$= 2 + \frac{11}{8}$$

$$= \frac{27}{8} \text{ bits}$$

Chain rule for relative entropy: $D(p(x,y) \| q(x,y)) = D(p(x) \| q(x)) + D(p(y|x) \| q(y|x))$

Proof: $D(p(x,y) \| q(x,y))$

$$= \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{q(x,y)}$$

$$= \sum_x \sum_y p(x,y) \log \frac{p(x)\,p(y|x)}{q(x)\,q(y|x)}$$

$$= \sum_x \sum_y p(x,y) \log \frac{p(x)}{q(x)} + \sum_x \sum_y p(x,y) \log \frac{p(y|x)}{q(y|x)}$$

$$= D(p(x) \| q(x)) + D(p(y|x) \| q(y|x))$$

Non-negativity of Mutual Information: $I(X;Y) \geq 0$

Proof: $I(X;Y) = D(p(x,y) \| p(x)p(y)) \geq 0$, with equality if and only if $p(x,y) = p(x)p(y)$

$\therefore D(p(y|x) \| q(y|x)) \geq 0$, with equality if and only if $p(y|x) = q(y|x)$ for all y and x such that $p(x) > 0$

Let (X,Y) have the following joint distribution:

$Y \backslash X$	1	2
1	0	$\frac{3}{4}$
2	$\frac{1}{8}$	$\frac{1}{8}$

$\Rightarrow$ The marginal distribution of X is $P(x_1) = \frac{1}{8}$, $P(x_2) = \frac{7}{8}$ and Y is $P(y_1) = \frac{3}{4}$, $P(y_2) = \frac{1}{4}$

$$\therefore H(X) = -\sum_{i=1}^{2} p(x) \log_2 p(x)$$

$$= -\frac{1}{8} \log_2 \frac{1}{8} - \frac{7}{8} \log_2 \frac{7}{8}$$

$$= 0.544 \text{ bits}$$

$$\therefore H(X|Y) = \sum_{i=1}^{2} P(Y=i)\, H(X|Y=i)$$

$$= P(Y=1)\, H(X|Y=1) + P(Y=2)\, H(X|Y=2)$$

$$= \frac{3}{4}H(0,1) + \frac{1}{4}H\left(\frac{1}{2},\frac{1}{2}\right)$$

$$= \frac{3}{4} \times (0+0) + \frac{1}{4}\left(\frac{1}{2}+\frac{1}{2}\right)$$

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

$$= 0.25 \text{ bits}$$

Uncertainity in X is increased if $Y=2$ is observed and decreased if $Y=1$ is observed, but uncertainity decreases on the average.

Log sum inequality: For non-negative numbers, $a_1, a_2 \ldots a_n$ and $b_1, b_2 \ldots b_n$

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} a_i\right) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}$$ with equality if and only if $\frac{a_i}{b_i} = \text{constant}$.

Proof: According to Jensen's inequality,

$$\sum \alpha_i f(t_i) \geq f\left(\sum \alpha_i t_i\right)$$

for $\alpha_i \geq 0$, $\sum_i \alpha_i = 1$. Setting $\alpha_i = \frac{b_i}{\sum_{j=1}^{n} b_j}$ and $t_i = \frac{a_i}{b_i}$

$$\sum \frac{a_i}{\sum b_j} \log \frac{a_i}{b_i} \geq \sum \frac{a_i}{\sum b_j} \log \sum \frac{a_i}{\sum b_j}$$

By the log sum inequality,

$$D(p\|q) = \sum p(x) \log \frac{p(x)}{q(x)}$$

$$\geq \left(\sum p(x)\right) \log \sum p(x) / \sum q(x)$$

$$= 1 \log \frac{1}{1} = 0$$

with equality if and only if $\frac{p(x)}{q(x)} = c$. Since both p and q are probability mass functions $c=1$. Hence we have $D(p\|q)=0$ if and only if $p(x)=q(x)$ for all x.

Stochastic Process: A stochastic process is ~~said~~ to an indexed sequence of random variables.

Markov Chain: A discrete stochastic process $X_1, X_2 \ldots$ is said to be a Markov chain or Markov process if for $n = 1, 2 \ldots$

$$Pr(X_{n+1} = x_{n+1} \mid X_n = x_n, X_{n-1} = x_{n-1} \ldots X_1 = x_1)$$

$$= Pr(X_{n+1} = x_{n+1} \mid X_n = x_n)$$

Source Coding: A source code C for a random variable X is a mapping from $\mathcal{X}$, the range of X, to D^*, the set of finite-length strings of symbols from a D-ary alphabet.

The source coding theorem states that the discrete memoryless source of entropy H(S), the average code-word length L for any distortionless source coding is bounded as

$$L \geq H(S)$$

This theorem provides the mathematical tool for assessing data compaction i.e. lossless data, compression of data generated by a discrete memoryless source.

Data Compression: Data compression means encoding information using fewer bits than original representation.

Data compression is two types: i) Lossy ii) Lossless

Lossy Compression: Lossy compression reduces the redundant information from the file.

Example – JPEG 2000

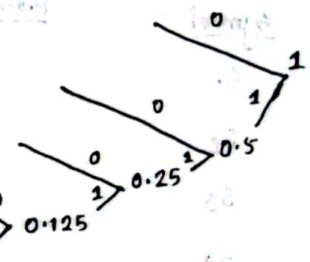
Lossless Compression: Lossless when every single bit of data that was originally in the file remains after the file is uncompressed.

Example – Huffman Coding, Arithmatic Coding, Runlength Coding

Consider a source with a symbol $X = \{a, b, c, d, e\}$ with probabilities $P = \{0.5, 0.25, 0.125, 0.0625, 0.0625\}$. Construct the Huffman Code.

Solution:

Symbol	Probabilities	Codewords
a	0.5	0
b	0.25	10
c	0.125	110
d	0.0625	1110
e	0.0625	1111



The average number of bits per symbol is

$$\text{average length} = 1 \times 0.5 + 2 \times 0.25 + 3 \times 0.125 + 4 \times 0.0625 + 4 \times 0.0625$$

$$= 1.875 \text{ bits/symbol}$$

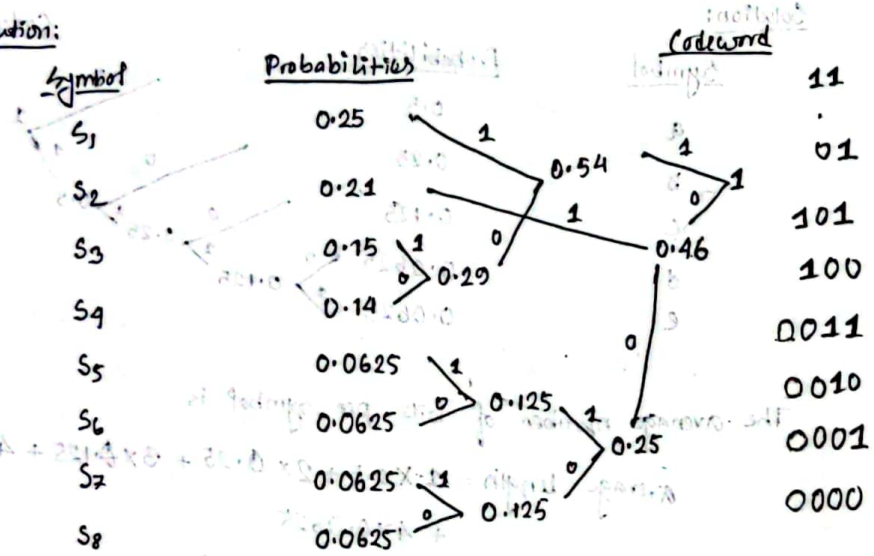
Entropy $H = \frac{1}{2}\log_2 \frac{1}{1/2} + \frac{1}{4}\log_2 \frac{1}{1/4} + \frac{1}{8}\log_2 \frac{1}{1/8} + \left(\frac{1}{16}\log_2 \frac{1}{1/16} \times 2\right)$

$$= 1.875 \text{ bits/symbol}$$

[Length = Codeword × probability]

Consider a source with eight symbols $S_1, S_2 \ldots S_7, S_8$ with probabilities 0.25, 0.21, 0.15, 0.14, 0.0625, 0.0625, 0.0625, 0.0625

Solution:



~~Codeword~~ ∴ Average length $= (0.25\times2)+(0.21\times2)+(0.15\times3)+(0.14\times3) + (4\times0.0625)\times4$

$= 2.79$

Entropy $H = 0.25\log_2\left(\frac{1}{0.25}\right) + 0.21\log_2\left(\frac{1}{0.21}\right) + 0.15\log_2\left(\frac{1}{0.15}\right) + 0.14\log_2\left(\frac{1}{0.14}\right) + 0.0625\log_2\left(\frac{1}{0.0625}\right)\times4$

$= 2.78$

$\therefore H(s) \le L(s)$

Fixed Length Coding and Variable Length Coding

Fixed Length Code

Letter	Codeword	Letter	Codeword
A	000	E	100
B	001	F	101
C	010	G	110
D	011	H	111

A variable length code for the same set of letters can be

Letter	Codeword	Letter	Codeword
A	00	E	101
B	010	F	110
C	011	G	1110
D	100	H	111

If we have to code the series of letters - "A BAD CAB"

Fixed length code : 000 001 000 011 010 000 001 = 21

Variable " " : 00 010 00 100 011 00 010 = 18

Variable length code 2:

Letter	Codeword	Letter	Codeword
A	0	E	10
B	1	F	11
C	00	G	000
D	01	H	111

Variable Length Code 1 : 00 010 00 100 011 00 010 = 18 bits

Variable Length Code 2 : 0 1001 0001 = 9 bits

Disadvantage → Decoding Problem

Advantage →
i) Data transfer rate high
ii) Less error
iii) Less memory
iv) Same data transfer

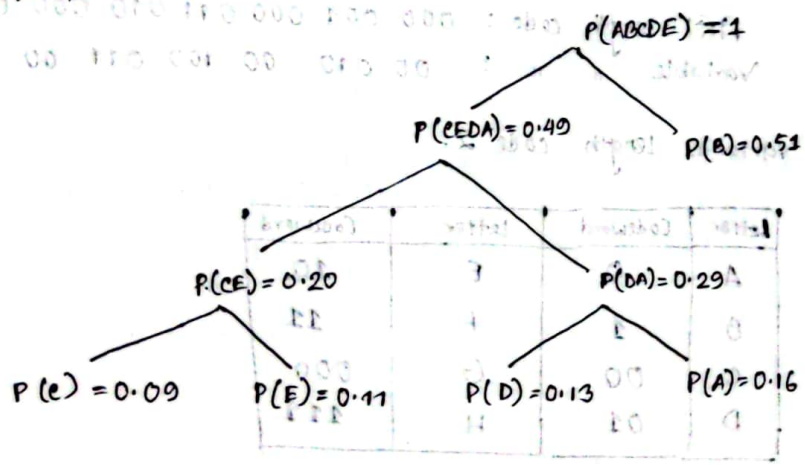
Solution of the decoding problem → Prefix Condition

Prefix Code: A prefix code is one in which no codeword forms the prefix of any other codeword.

Huffman Coding (Method 2)

Given, $P(A) = 0.16$, $P(B) = 0.51$, $P(C) = 0.09$, $P(D) = 0.13$, $P(E) = 0.11$

⇒



Kraft Inequality:

For any uniquely decodable code,

$$\sum_{k=1}^{L} 2^{-n_k} \le 1 \quad , \; n = \text{length of bit where } n_1 \le n_2 \le n_3 \le \dots \le n_L$$

Kraft inequality doesn't tell us whether the code is prefix free or not.

When certain length satisfy Kraft inequality, it is possible to construct a prefix code in which the codewords are of these lengths. This implies there exist non prefix codes with these lengths. Non-prefix codes can also be constructed with these lengths. Thus the code doesn't necessarily have to be a prefix code if its codeword lengths satisfy the Kraft inequality.

For example, code (0, 01, 001, 010) satisfies the Kraft inequality since $K(c) = \sum_{j=1}^{n} 2^{-l_j} = \frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^3} = 1.$

However, it is obviously not a prefix code because the first codeword 0 is the prefix of the others.

Coding Efficiency:

$$\eta = \frac{\text{Minimum bit length}}{\text{Average Codeword length}} = \frac{L_{min}}{L_a}$$

From Shannon's Theorem,

$$L_a \geq H(s)$$

Thus, $L_{min} = H(s)$

Hence, $\eta = \frac{H}{L}$.

Given minimum bit length 1.846 and Average codeword length is 2. Calculate the coding efficiency.

⇒ We know,

$$\text{Coding efficiency, } \eta = \frac{H}{L}$$

$$= \frac{1.846}{2}$$

$$= 0.923$$

$$= 92.3\%$$

Arithmetic Coding: Arithmetic coding is a type of entropy encoding utilized in lossless data compression. It is one kind of variable length coding.

Arithmetic coding converts a variable number of symbols into variable length codeword. It represents group of symbol together.

Upper limit = Lower limit + Difference × Probability

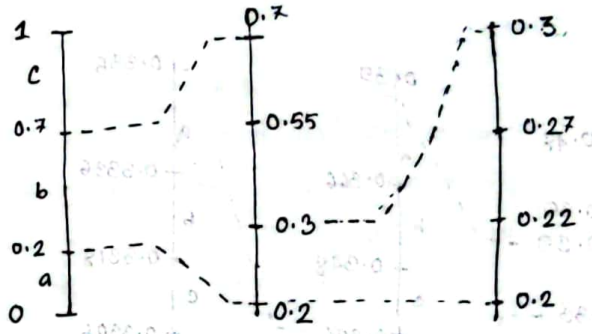
Advantages:

1. Avoids the disadvantage of Huffman coding
2. It encodes a whole string of symbols in one floating point number.
3. Redundancy much reduced

Given that, $A = \{a, b, c\}$, $P(a) = 0.2$, $P(b) = 0.5$, $P(c) = 0.3$.

Given sequence, bac

Solution:



For b, $a = 0.2 + 0.5 \times 0.2 = 0.3$

$b = 0.3 + 0.5 \times 0.5 = 0.55$

$c = 0.55 + 0.5 \times 0.3 = 0.7$

For a, $a = 0.2 + 0.1 \times 0.2 = 0.22$

$b = 0.22 + 0.1 \times 0.5 = 0.27$

$c = 0.27 + 0.1 \times 0.3 = 0.3$

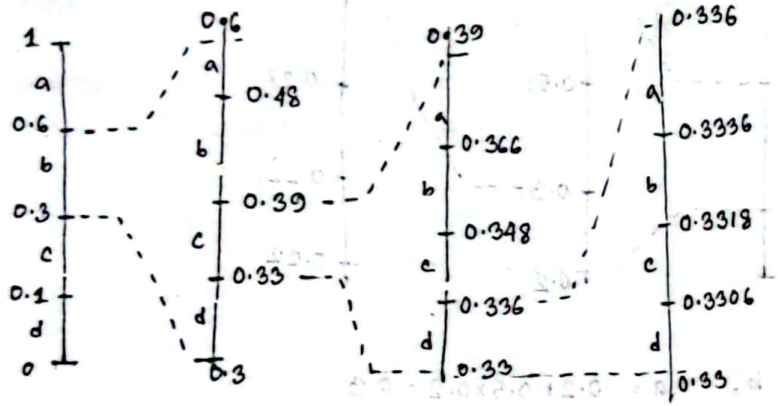
$$\text{Tag} = \frac{0.3 + 0.27}{2} = 0.285$$

$$(0.285)_{10} = (0.01001\ldots)_2$$

$$\begin{array}{r} .285 \\ \times\ 2 \\ \hline 0.57 \\ \times\ 2 \\ \hline 1.14 \\ \times\ 2 \\ \hline 0.28 \\ \times\ 2 \\ \hline 0.56 \\ \times\ 2 \\ \hline 1.12 \end{array}$$

Given that, $P(a)=0.4$, $P(b)=0.3$, $P(c)=0.2$, $P(d)=0.1$ and sequence is beda.

Solution:



For b, $d = 0.3 + 0.3 \times 0.1 = 0.33$

$e = 0.33 + 0.3 \times 0.2 = 0.39$

$b = 0.39 + 0.3 \times 0.3 = 0.48$

$a = 0.48 + 0.3 \times 0.4 = 0.6$

For e, $d = 0.33 + 0.06 \times 0.1 = 0.336$

$e = 0.336 + 0.06 \times 0.2 = 0.348$

$b = 0.348 + 0.06 \times 0.3 = 0.366$

$a = 0.366 + 0.06 \times 0.4 = 0.39$

For d, $d = 0.33 + 0.006 \times 0.1 = 0.3306$

$e = 0.3306 + 0.006 \times 0.2 = 0.3318$

$b = 0.3318 + 0.006 \times 0.3 = 0.3336$

$a = 0.3336 + 0.006 \times 0.4 = 0.336$

$(0.3336)_{10} = (0.01010 \quad)_2$

Lampel Ziv Encoding: A compression algorithm that codes string of characters with codes of a fixed number of bits.

Advantages:

1. Fast and simple
2. Most popular lossless compression
3. Reduces file size containing repetitive data
4. No need to pass string table to decompression
5. Table can be recreated

Process:

1. Accomplished by diving data into segments
2. These sequence are shortest and not encountered previously
3. All segments retreated as phrase
4. Encode phrase by prefix and last bit

Numerical representation = Prefix Index Position + Last bit

Shorest symbol - Non repetitive

Method 1:

1	2	3	4	5	6	7
B	A	A B	A B B	B A	A B B B	B A A
∅B	∅A	2B	3B	1A	4B	5A
∅1	∅0	21	31	10	41	50

A = 0
B = 1

⇒ 01 00 0101 0111 10 1001 1010

1	0	10	11	01	101	010	1011
1	2	3	4	5	6	7	8

Numerical Position	1	2	3	4	5	6	7	8
Subsequence/ Index	1	0	10	11	01	101	010	1011
Numerical Representation	–	–	10	11	21	31	50	61
Binary encoded Blocks	0001	0000	0010	0011	0101	0111	1010	1101

Method 2:

bacacababacabacabavariable indexing (1) a (2) b (3) c

⇒

Read	Produce	Write
		(1) a
		(2) b
		(3) c
b		(4) ba
a	(2)	(5) ac
c	(1)	(6) ca
a	(3)	(7) aca
c	(5)	(8) ab
a	(1)	(9) bab
b	(4)	
a		
b		
a		
b		

Read	Produce	Write
a	(4)	(10) baba
c	(7)	(11) acab
a	(1)	a
b		
a		

Encoded output: 21351447 1

ABABBABCABABBA

⇒

Read	Produce	Write
		(1) A
		(2) B
		(3) C
A		
B	(1)	(4) AB
A	(2)	(5) BA
B		
B	(4)	(6) ABB
A		(7) BAB
B	(5)	
C	(2)	(8) BC
A	(3)	(9) CA
B	(4)	(10) ABA
A		
B	(6)	(11) ABBA
B		
A		

Encoded Output = 1245 23 4C

Decoding:

Given sequence (3) (1) (2) (5) (1) (4) (6) (6)

Read	Produce	Write
		(1) a
		(2) b
		(3) c
(3)	c	
(1)	a	(4) ca
(2)	b	(5) ab
(5)	ab	(6) ba
(1)	a	(7) aba
(4)	ca	(8) ac
(6)	ba	(9) cab
(6)	ba	(10) bab

Decoded Output : cababacababa

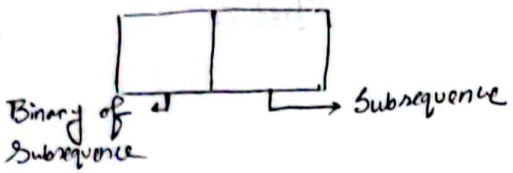
Given Sequence : (1) (2) (4) (5) (2) (3) (4) (6) (1)

→

Read	Produce	Write
		(1) A
		(2) B
		(3) C
(1)	A	
(2)	B	(4) AB
(4)	AB	(5) BA
(5)	BA	(6) ABB
(2)	B	(7) BAB
(3)	C	(8) BC
(4)	AB	(9) CA
(6)	ABB	(10) ABA
(1)	A	(11) ABBA

Decoded Output : ABABBABCABABBA

Binary Encoded Block:



$$R(D) = P(\hat{x}|x) : E[d(x,\hat{x})] \leq D$$

$$d(x_k, \hat{x}_k) = (x_k - \hat{x}_k)^2$$

$$d(X_n, \hat{X}_n) = \frac{1}{n}\sum_{k=1}^{n} d(x_k, \hat{x}_k)$$

$$D = E[d(X_n, \hat{X}_n)] = \frac{1}{n}\sum_{k=1}^{n} E[d(x_k, \hat{x}_k)]$$

Average number of bit per sample is required for representing a given source without exceeding a given distribution

Bounding factor → Lossy - Rate of distortion
Lossless - Entropy

Run Length Encoding:

- Run length encoding (RLE) is a very simple form of lossless data compression in which runs of data are stored as a single data value than as the original run.

- The process involves going through the text and counting the number of consecutive occurences of each character (called 'a run'). The number of occurences of the characters and the character itself are then stored in pairs.

Advantages and Disadvantages:

Advantages:

1. It is very easy to implement.
2. It doesn't require much CPU memory.
3. Works best for images with solid background.

Disadvantages:

1. It doesn't work well for natural image
2. It is only efficient with files that contain lots of repetitive data.
3. It doesn't guarantee space savings
4. It must have consecutive runs of values in order to maximize compression.

CamScanner

Example:

- Original Data: BBBBBBAAAAPPPQQQSWWWWTTTTTTT
  6 4 3 3 1 4 7

Compressed Data: B06A04P03Q03S01W04T07

Original Data Length: 28

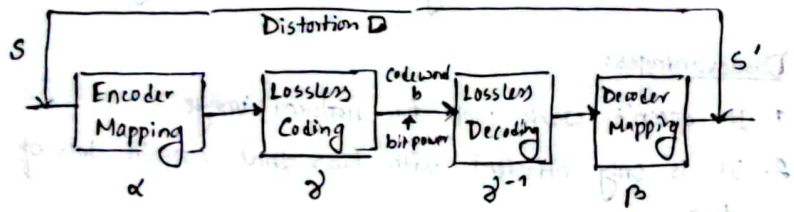
Compressed Data Length: 21

- Original data: Code for 0 only

0000000000000010000110000000000000 (33)
14 4 8 12

Compressed Data: 1110 0100 0000 1100 (16)

- Operational Rate Distortion:



Hamming distance: It is the number of symbols or positions of two strings at which their corresponding characters are different.

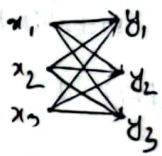
- Channel Coding:

In the transmitter, before a message is sent, it's encoded. Then in the receiver, it can be decoded to recover the original message and remove all of the noise, interference and effects of poor signal strength. This is called channel coding.

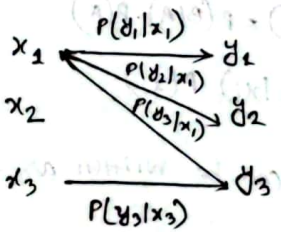
- Discrete Memoryless Channel: Input and output → finite set of symbols

For ideal channel, input = output ; $x_1 = y_1$, $x_2 = y_2$ ... $x_n = y_n$

For non-ideal channel,



-



A discrete memoryless channel is characterized by an input range of symbols $\{x_1, x_2 \ldots x_i\}$ and output range of $\{y_1, y_2 \ldots y_j\}$ and a set of conditional probabilities

$P(y_j|x_i)$ which determines the relationship between the input x_i and the output y_j.

The transitional probability of the channel P is given by

$$P = \begin{bmatrix} P(y_1|x_1) & P(y_2|x_1) & P(y_3|x_1) \\ P(y_1|x_2) & P(y_2|x_2) & P(y_3|x_2) \\ P(y_1|x_3) & P(y_2|x_3) & P(y_3|x_3) \end{bmatrix}$$

i) Each row of P contains the probabilies of all possible outputs from the same input to the channel.

ii) Each column of P contains the probabilities of all possible inputs to a particular output from the channel

iii) If we transmit the symbol x_i, we must receive an output symbol with probability 1.

That is, $\sum_{j=1}^{J} P(y_j|x_i) = 1$

From probability theory $P(A,B) = P(B|A) \cdot P(A)$

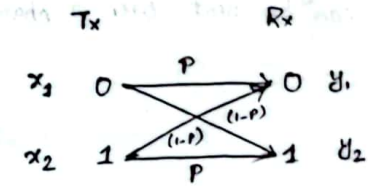
Therefore, $P(x_i, y_j) = P(y_j|x_i) \cdot P(x_i)$

Hence the Channel matrix can be written as -

$$P(y_j|x_i) \cdot P(x_i) = \begin{bmatrix} P(y_1|x_1) \cdot P(x_1) & P(y_2|x_1) \cdot P(x_1) & \cdots & P(y_j|x_1) \cdot P(x_1) \\ P(y_1|x_2) \cdot P(x_2) & P(y_2|x_2) \cdot P(x_2) & \cdots & P(y_j|x_2) \cdot P(x_2) \\ \vdots & & & \\ P(y_1|x_i) \cdot P(x_i) & P(y_2|x_i) \cdot P(x_i) & \cdots & P(y_j|x_n) \cdot P(x_n) \end{bmatrix}$$

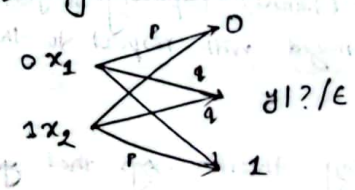
$$= P(x_i, y_j) = \begin{array}{c} \\ i=1 \\ i=2 \\ \vdots \\ i=n \end{array} \begin{array}{c} \begin{array}{cccc} j=1 & j=2 & \cdots & j=m \end{array} \\ \begin{bmatrix} P(x_1,y_1) & P(x_1,y_2) & \cdots & P(x_1,y_m) \\ P(x_2,y_1) & P(x_2,y_2) & \cdots & P(x_2,y_m) \\ \vdots & & & \\ P(x_n,y_1) & P(x_n,y_2) & \cdots & P(x_n,y_m) \end{bmatrix} \end{array}$$

Binary Symmetric Channel:



$$P = \begin{bmatrix} P(y_1|x_1) & P(y_2|x_1) \\ P(y_1|x_2) & P(y_2|x_2) \end{bmatrix} = \begin{bmatrix} P & 1-P \\ 1-P & P \end{bmatrix}$$

Binary Erasure Channel:



$$P = \begin{bmatrix} P(y=0|x=0) & P(y=y|x=0) & P(y=1|x=0) \\ P(y=0|x=1) & P(y=y|x=1) & P(y=1|x=1) \end{bmatrix}$$

$$= \begin{bmatrix} P & q & 0 \\ 0 & q & P \end{bmatrix}$$

Let, a DMC with input x and output Y and transition probability $P(y_j|x_i)$, then mutual information of the channel

$$I(x;Y) = \sum_{i=0}^{j-1} \sum_{j=0}^{j-1} P(x_i) \cdot P(y_j|x_i) \cdot \log \frac{P(y_j|x_i)}{P(y_j)}$$

Channel capacity of a DMC is the maximum average mutual information where maximum is overall possible input probabilities.

Channel Capacity: Channel capacity C, measures the maximum amount of information that can be sent over a channel. It is defined as follows:

$$C = \max_{P(x_i)} I(X,Y)$$

where X is the input to the channel and Y is the output.

Because errors occur in transmission, the relationship between X and Y will not be completely deterministic. Mutual information measures how much information about X gets through the channel, but it will depend on the input distribution $P(x)$. Channel capacity is just the mutual information maximized with respect to the input distribution $P(x)$

Shannon's noisy channel coding theorem says that you can transmit information which arbitrarily small probability of error at a rate R which is less than the channel capacity C.

Thus if you wish to achieve error free communication, the source entropy $H(x)$ must be less than the channel capacity C. Error free communication above the channel capacity is impossible.

Derive $C = 1 - H_b(P)$

$\Rightarrow$ We know, capacity of DMC,

$$C = \max_{P(x_i)} I(x;Y)$$

$$= \max_{P(x_i)} \sum_{j=0}^{q-1} \sum_{i=0}^{p-1} P(x_i) \cdot P(y_j|x_i) \cdot \log \frac{P(y_j|x_i)}{P(y_j)} \quad \text{--- (1)}$$

The maximum of $I(x;Y)$ is performed when

$$P(x_i) \geq 0 \text{ and } \sum_{i=0}^{q-1} P(x_i) = 1$$

Now, Let a Binary Symmetric Channel with transition probabilities

$$P(0|1) = P = P(1|0)$$

From (1),

$$C = 1 + P\log_2 P + (1-P)\log_2(1-P) \quad \text{--- (2)}$$

As the entropy function,

$$H_b(P) = -P\log_2 P - (1-P)\log_2(1-P) \quad \text{--- (3)}$$

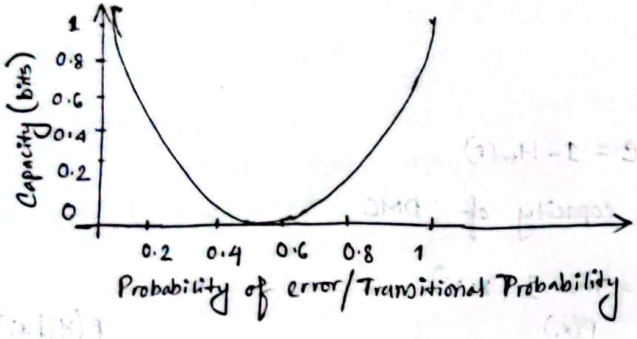
Putting (3) to (2),

$$C = 1 - H_b(P)$$

CamScanner

The capacity of the binary symmetric channel

⇒ The capacity of the binary symmetric channel can be written as $C = 1 - H_b(P)$

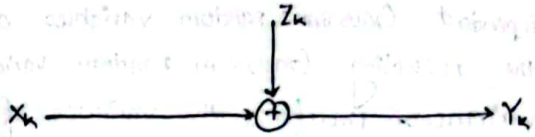


When $P = 0.5$, the capacity diminishes to 0. This is because we have an equal probability of successful and unsuccessful transmission. In other words, we have no idea whether the transmitted bit was flipped. On the other hand, as $P \to 1$ or $P \to 0$, we have 100% certainity that the transmission was successful or 100% certainity that the transmitted bit was flipped. That is, we know with absolute certainity what the transmitted symbol (bit) was. This makes the channel capacity $C = 1$ bit per channel use.

Gaussian Channel:

A Gaussian channel is a discrete time channel with output Y_k, input X_k and noise Z_k at time k such that $Y_k = X_k + Z_k$, $Z_k \sim N(0, N)$ where Z_k is independent of X_k.

This noise is drawn from a Gaussian distribution with mean zero and variance σ^2



Capacity of Gaussian Channel:

Since the transmitter is usually power-limited. Let us put a constraint on the average power in X_k.

$$E[X_k^2] = P \qquad k = 1, 2, \ldots K$$

Thus, the information capacity of the channel is given by

$$C = \max_{f_{X_k}(x)} \{ I(X;Y) \mid E[X_k^2] = P \}$$

$$I(X_k; Y_k) = h(Y_k) - h(Y_k \mid X_k)$$

X_k and N_k are independent random variables. Therefore, the conditional differential entropy of Y_k given X_k is equal to the differential entropy of N_k. Intuitively, this is because given the uncertainity arising in Y_k is purely due to N_k. That i

$$h(Y_k \mid X_k) = h(N_k)$$

Hence, we can write,

$$I(X_k; Y_k) = h(Y_k) - h(N_k)$$

In order to maximize the mutual information between the input X_k and the channel output Y_k, the transmitted signal should be Gaussian. We can write,

$$C = I(X;Y) \mid E[X_k^2] = P \quad \text{and} \quad X_k \text{ is Gaussian}$$

CamScanner

If two independent Gaussian random variables are added, the variance of the resulting Gaussian random variable is the sum of the variances. Therefore, the variance of the received sample $Y_k = P + N_0 W$

It can be shown that the differential entropy of a Gaussian random variable with variance σ^2 is $\frac{1}{2}\log_2(2\pi e\sigma^2)$

Therefore, $h(Y_k) = \frac{1}{2}\log_2[2\pi e(P+N_0W)]$

and $h(N_k) = \frac{1}{2}\log_2[2\pi e(N_0W)]$

$$\therefore \quad C = h(Y_k) - h(N_k)$$

$$= \frac{1}{2}\log_2[2\pi e(P+N_0W)] - \frac{1}{2}\log_2[2\pi e(N_0W)]$$

$$= \frac{1}{2}\log_2\left(1+\frac{P}{N_0W}\right) \text{ bits per channel use.}$$

If we transmit $2W$ samples per second, the information capacity can be expressed as -

$$C = W\log_2\left(1+\frac{P}{N_0W}\right) \text{ bits per second}$$

It is known as the Shannon's third theorem, the Information Capacity Theorem.

Let us define an ideal system which can transmit data at a bit rate R_b which is equal to the capacity C of the channel. i.e. $R_b = C$.

Then average transmitted power is $P = E_bR_b$
$= E_bC$

From Shannon's third theorem,

$$C = W\log_2\left(1+\frac{P}{N_0W}\right)$$

$$\Rightarrow \frac{C}{W} = \log_2\left(1+\frac{E_bC}{N_0W}\right)$$

$$\Rightarrow 2^{C/W} = 1 + \frac{E_b}{N_0}\cdot\frac{C}{W}$$

$$\Rightarrow 2^{C/W} - 1 = \frac{E_b}{N_0}\cdot\frac{C}{W}$$

$$\therefore \frac{E_b}{N_0} = \frac{2^{C/W}-1}{C/W}$$

When $C/W = 1$,

$$\frac{E_b}{N_0} = \frac{2^1-1}{1} = 1$$

When $C/W \to \infty$

$$\frac{E_b}{N_0} \approx \frac{2^{C/W}}{C/W} \approx \exp\left(\frac{C}{W}\ln 2 - \ln\frac{C}{W}\right)$$

For infinite bandwidth, $W \to \infty$

$$\left.\frac{E_b}{N_0}\right|_{W\to\infty} = \frac{2^{C/W}\ln 2 - 0}{1} \quad \text{[Taking La hospital]}$$

$$= \ln 2$$

$$= 0.693$$

$$= 10\log 0.693 = -1.6 \text{ dB}$$

CamScanner

Shannon Channel Capacity Theorem

The Shannon channel capacity theorem defines the theoretical maximum bit rate (number of bits per second) for a noisy channel

$$\text{Capacity, } C = B\log_2\left(1+\frac{S}{N}\right)$$

Calculate the bit rate for a noisy channel with SNR 300 and bandwidth of 3000 Hz.

Solution: The bit rate for a noisy channel,

$$C = B\log_2\left(1+\frac{S}{N}\right)$$
$$= 3000\log_2(1+300)$$
$$= 24700 \text{ bps}$$

Consider an extremely noisy channel in which the value of signal to noise ratio is almost zero. What is the capacity of the channel?

Solution: The bit rate for a noisy channel,

$$C = B\log_2(1+S/N)$$
$$= B\log_2(1+0)$$
$$= B\log_2 1$$
$$= 0$$

We can't receive any data through this channel.

A telephone line normally has a bandwidth of 3000 Hz (300 to 3300 Hz) assigned for data communications. The signal-to-noise ratio is usually 3162. Calculate the highest bit rate.

Solution: The bit rate for a noisy channel,

$$C = B\log_2(1+SNR)$$
$$= 3000\log_2(1+3162)$$
$$= 34881 \text{ bps.}$$

Shannon's Capacity Formula

If the data rate of the channel is less than the error-free capacity, then it is theoretically possible to use a suitable code to achieve error free transmission through the channel.

The data rate could be increased by increasing either the signal strength or the bandwidth. However -

i) Increasing the bandwidth, increments the costs

ii) Increasing the signal strength, increments the effect of non-linearities in the system producing an increase in inter-modulation noise.

Advantages:

1. The information capacity theorem provides a bound on what rate of transmission is theoritically attainable for prescribed values of chan bandwidth B and received SNR.

2. It provides an idealized framework for comparing the noise performa of one modulation scheme against another.

CamScanner

- ARQ (Automatic Repeat Request): Error detection - Full duplex

rcvr → ACK / NACK → transmitter
↓
if error found

- Forward Error Correction (FEC): Simpler
- Hybrid HRQ: Full duplex → Error Detection + Correction

Data	Parity	Data with Parity	Check
1	1	11	Even
1010	1	10101	Odd
1010	0	10100	Even
111111	1	1111111	Odd

Parity Check → Number of 1

Character	Sender	Parity bit	Receiver	Parity Check
E	1000101	1	10001011	Even
A	1000001	0	10000010	Even
'q'	1110001	0	11100000	Odd

- Drawbacks:

i) It doesn't tell us about the bit flip

ii) If bit flip occurs, it can't detect the error. Thus, we can't ensure properly received bit

- Channel encoder provide redundent bit

Symbol (k bit) $\xrightarrow{\text{Redundent Data, r}}$ Channel

∴ Codeword, $n = k + r$

- If $n = 010\,010$, $k = 010$

∴ Encoder efficiency, $\eta = \frac{k}{n} = \frac{3}{6} = 0.5$

- Hamming Distance: The hamming distance between two codeword is equal to the number of difference between them.

d(10101, 11110)

1 0 1 0 1 → Codeword 1
1 1 1 1 0 → Codeword 2

∴ Hamming distance = 3

Alternatively, we can compute by adding codewords. Therefore, by X-OR operation and counting the number of ones.

$$\begin{array}{r} 10101 \\ \oplus\ 11110 \\ \hline 01011 \end{array}$$

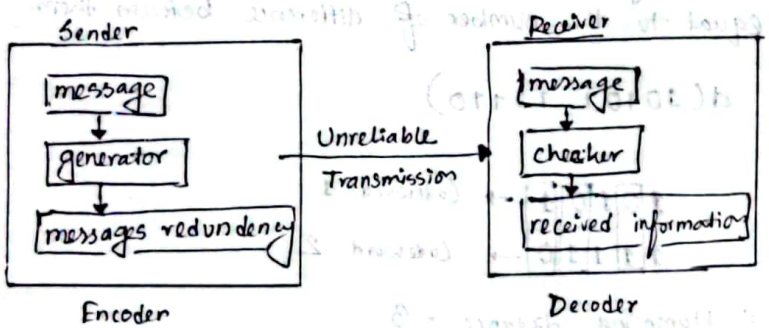
∴ Hamming distance = 3

CamScanner

Hamming Weight: Hamming weight of the codeword is defined as the non-zero components in a word.

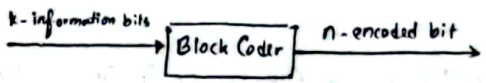
Codewords	Weight
000000	0
001110	3
010101	3
110101	4

The Structure of encoder and decoder



Linear Block Code:

In block coding, the binary information sequence is segmented into message blocks of fixed length.



n-digit codeword made up of k-information and (n-k) redundant parity check digits. The rate for this code is k/n.

$$\text{Efficiency}, \; r = \frac{k}{n}$$

If message block consists of k information digit is u and distinct message is 2^k, each input message u is transform into binary n-tuple v with $n>k$. The binary n-tuple v is referred to as codeword of message u.

For 2^k message, 2^k codewords are possible. Set of 2^k code are called block code.

Block Code: A block code of length n and 2^k codewords called a linear code (n,k) if and only if 2^k codew form a k-dimensional subspace of the vector space of the n-tuples over the field GF

Linear transformation of binary block code:

$$C = m.G$$

$$= \sum_{i=0}^{k-1} m_i g_i = m_0 g_0 + m_1 g_1 + \dots + m_{k-1} g_{k-1}$$

Systematic Code: A systematic block code consists of vector whose 1st k elements (or last k-elements) are identical to the message bits, the remaining (n-k) elements being check bits.

$$x = (m_0, m_1, m_2 \ldots, m_{k-1}, c_0, c_1, c_2 \ldots c_{n-k})$$

or,

$$x = (c_0, c_1, c_2 \ldots c_{n-k}, m_0, m_1, m_2 \ldots m_{k-1})$$

Systematic Code: Information digits are explicitly transmitted together with the parity check bits. For the code to be systematic, the k-information bits must be transmitted contiguously as a block, with the parity check bits making up the code word as another contiguous block.

Information bits	Parity bits

A systematic linear block code will have a generator matrix of the form:

$$G = [P | I_k]$$

Parity Check Matrix (H): Parity check matrix will enable us to decode the received vectors. For each $(k \times n)$ generator matrix G, there exists an $(n-k) \times n$ matrix H, such that rows of G are orthogonal to rows of H. i.e. $GH^T = 0$. To fulfill the orthogonal requirements for a systematic code, the components of H matrix are written as

$$H = [I_{n-k} | P^T]$$

Codeword: The codeword C is compromised of an information segment and a set of $(n-k)$ symbol that are linear combination of certain information symbol.

A codeword consists of n digits $x_0, x_1, x_2 \ldots x_{n-1}$ and a data word (message word) consists of k digits $m_0, m_1, m_2 \ldots m_{k-1}$

$$c = (m_0, m_1, m_2 \ldots m_{k-1}, c_k, c_{k+1} \ldots c_{n-1})$$

Linear Property: Linear means sum of any 2 codewords yields another codeword. A binary code is linear if and only if the modulo-2 sum of 2 codewords is also a codeword.

The generator matrix for a (6,3) block code is given below. Find all the code vectors of this code. $G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$

Solution: $(n, k) = (6, 3)$

$\therefore n = 6, \quad k = 3, \quad (n-k) = (6-3) = 3$ number of parity bits

We know, $G = [I_k \mid P]$

$$\therefore I_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad P = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Code Vector. $C = MP$

$$[c_0 \; c_1 \; c_2] = [m_0 \; m_1 \; m_2] \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$c_0 = (m_0 \cdot 0) \oplus (m_1 \cdot 1) \oplus (m_2 \oplus 1)$ $\qquad c_1 = m_0 \oplus m_2$

$= m_1 \oplus m_2$ $\qquad c_2 = m_0 \oplus m_1$

For message word $(m_0 m_1 m_2 = 000)$

$C_0 = 0$, $C_1 = 0$, $C_2 = 0$

For message word $(m_0 m_1 m_2 = 001)$

$C_0 = 1$, $C_1 = 1$, $C_2 = 0$

For message word $(m_0 m_1 m_2 = 010)$

$C_0 = 1$, $C_1 = 0$, $C_2 = 1$

For message word $(m_0 m_1 m_2 = 011)$

$C_0 = 0$, $C_1 = 1$, $C_2 = 1$

For message word $(m_0 m_1 m_2 = 100)$

$C_0 = 0$, $C_1 = 1$, $C_2 = 1$

For message word $(m_0 m_1 m_2 = 101)$

$C_0 = 1$, $C_1 = 0$, $C_2 = 1$

For message word $(m_0 m_1 m_2 = 110)$

$C_0 = 1$, $C_1 = 1$, $C_2 = 0$

For message word $(m_0 m_1 m_2 = 111)$

$C_0 = 0$, $C_1 = 0$, $C_2 = 0$

$C_0 = m_1 \oplus m_2$

$C_1 = m_0 \oplus m_2$

$C_2 = m_0 \oplus m_1$

Code Vectors for (6,3) linear block code

S. No	Message Vector	Parity bits	Code Vectors
1	000	000	000000
2	001	110	001110
3	010	101	010101
4	011	011	011011
5	100	011	100011
6	101	101	101101
7	110	110	110110
8	111	000	111000

Linearity Check:

Codeword 2 : 001110

$\oplus$ Codeword 3 : 010101

011011

This codeword exists in codeword 4.

Codeword 4 : 011011

$\oplus$ Codeword 7 : 110110

101101

This codeword exists in codeword 6.

CamScanner

The parity check matrix of a particular (7,4) linear block code is given by

$$H = \left[\begin{array}{cccc|ccc} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$

i) Find the generator matrix.

ii) List all the codewords

iii) What is the minimum distance between the code vectors?

iv) How many errors can be detected? How many errors can be corrected?

Solution: We know,

$$H = [P^T \mid I_{n-k}]$$

where P^T is $(n-k)\times k$ matrix and I_{n-k} is $(n-k)\times(n-k)$ matrix

$(n,k) = (7,4)$ $\therefore$ $n = 7, k = 4$, $n-k = (7-4) = 3$

$$\therefore P^T = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}_{3\times 4} \qquad I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}_{3\times 3}$$

Generator Matrix, $G = [I_k \mid P]$

$$\therefore P = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad \therefore G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

iii) Parity vector, $C = MP$

$$= [C_0 \quad C_1 \quad C_2] \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

$C_0 = m_0 \oplus m_1 \oplus m_2$, $C_1 = m_0 \oplus m_1 \oplus m_3$, $C_2 = m_0 \oplus m_2 \oplus m_3$

As the size of message block $k = 4$, there are $2^4 = 16$ possible message sequence

The complete code of (7,4) linear code is given below-

S. No.	Message bits	Parity bits	Codeword	Weight
1	0000	000	0000000	0
2	0001	011	0001011	3
3	0010	101	0010101	3
4	0011	110	0011110	4
5	0100	110	0100110	3
6	0101	101	0101101	4
7	0110	011	0110011	4
8	0111	000	0111000	3
9	1000	111	1000111	4
10	1001	100	1001100	3
11	1010	010	1010010	3
12	1011	001	1011001	4
13	1100	001	1100001	3
14	1101	010	1101010	4
15	1110	100	1110100	4
16	111	111	1111111	7

CamScanner

(iii) Minimum distance, d_{min} = Minimum weight of any non zero code vector

$= 3$

(iv) Number of errors can be detected

$d_{min} \geq s+1$

$\Rightarrow 3 \geq s+1$

$\therefore s \leq 2$

$\therefore$ Maximum error can be detected $= 2$

Number of errors that can be corrected is given by

$d_{min} \geq 2t+1$

$\Rightarrow 3 \geq 2t+1$

$\therefore t \leq 1$

$\therefore$ Maximum errors that can be corrected $= 1$

$\therefore$ For (7,4) linear block code, at the most 2 errors can be detected and only 1 error can be corrected.

$$G = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

, $u = (1101)$ is the message to be encoded.

$\Rightarrow$ Codeword, $c = uG$

$$= \begin{bmatrix} 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

(7,4) hamming code over GF(2). The encoding equation for the code is-

$c_0 = m_0$, $c_1 = m_1$, $c_2 = m_2$, $c_3 = m_3$, $c_4 = m_0 + m_1 + m_2$,

$c_5 = m_1 + m_2 + m_3$, $c_6 = m_0 + m_1 + m_3$

Find out the generator matrix.

<u>Solution:</u>

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

CamScanner

Given that, $G = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$, $m = (1011)$ is the message to be encoded.

Solution: Equations for codewords will be -

$c_0 = m_0 + m_2 + m_3$

$c_1 = m_0 + m_1 + m_2$

$c_2 = m_1 + m_2 + m_3$

$c_3 = m_0$

$c_4 = m_1$

$c_5 = m_2$

$c_6 = m_3$

If message is $m = (1011)$

Then,

$c_0 = 1 + 1 + 1 = 1$

$c_1 = 1 + 0 + 1 = 0$

$c_2 = 0 + 1 + 1 = 0$

$c_3 = 1$ $\quad c_5 = 1$

$c_4 = 0$ $\quad c_6 = 1$

$\therefore$ Then codewond $= (1001011)$

Consider the code $c = \{0000, 1010, 0101, 1111\}$. The received word is $v = 1101$. Find the valid codeword of v.

Solution: Here C is a linear block code of length $n = 4$. This is a (4,2) code. Observe that all the ten possible sum of the codeword -

$0000 + 0000 = 0000$

$0000 + 1010 = 1010$

$0000 + 0101 = 0101$

$0000 + 1111 = 1111$

$1010 + 1010 = 0000$

$1010 + 0101 = 1111$

$1010 + 1111 = 0101$

$0101 + 0101 = 0000$

$0101 + 1111 = 1010$

$1111 + 1111 = 0000$

$d(1101, 0000) = 3$, $d(1101, 1011) = 2$, $d(1101, 0101) = 1$, $d(1101, 1110) = 2$

Vector $e = 1000$, the coset leader. Estimated Codeword 0101

Syndrome Decoding:

Let the transmitted vector be 'x' and corresponding received code vector be 'Y'. Then we can write -

$x = Y$ if there are no transmission errors

and $x \neq Y$ if there are errors created during transmission.

The decoder detects or corrects those errors in 'Y' using the stored bit pattern the decoder about the code. For larger block length more and more bits are required to be stored in the decoder. This increases the memory requirement and adds complexity a cost of the system. To avoid these problems, syndrome decoding used in linear block codes.

To minimize the probability of a decoding error it is des. to design a code, which corrects the 2^{n-k} most probable err pattern.

The $(n-k)$ vector S is defined as -

$$S = \underline{r} H^T$$
$$= (v + e) H^T$$
$$= e H^T \qquad [v \cdot H^T = 0]$$

This equation gives a relationship between unknown error and the syndrome.

7] The parity check matrix of a (7,4) Hamming code is as under. Calculate the syndrome vector for single bit.

$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Solution:

Syndrome Vector, $[S] = [E]_{1\times7} [H^T]_{7\times3}$

$\therefore$ Size of syndrome vector $= 1\times3 = [S]_{1\times3}$

$$H^T = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

To find syndrome corresponding to each error vector–

$$[S_1] = EH^T = [1000000] \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= [1\oplus0\oplus0\oplus0\oplus0\oplus0\oplus0 \quad 1\oplus0\oplus0\oplus0\oplus0\oplus0\oplus0 \quad 1\oplus0\oplus0\oplus0\oplus0\oplus0\oplus0]$$

$$= [111]$$

Similarly, $[S_2] = [1\ 0\ 0]$, $[S_3] = [1\ 0\ 1]$, $[S_4] = [0\ 1\ 1]$

$[S_5] = [1\ 0\ 0]$, $[S_6] = [0\ 1\ 0]$, $[S_7] = [0\ 1\ 1]$

Various code errors with single bit errors are shown below–

S. No	Error Vector	Bit in Error	Syndrome Vector
1	1000000	1st	111
2	0100000	2nd	100
3	0010000	3rd	110
4	0001000	4th	101
5	0000100	5th	011
6	0000010	6th	100
7	0000001	7th	011

To get code efficiency $r \geq 0.8$, we need $q \geq 5$

Then, $r = \frac{k}{n}$

$q = 5$, $n = 2^q - 1 = 2^5 - 1 = 31$

$\therefore k = 2^q - q - 1 = 31 - 5 - 1 = 26$

$\therefore 2^k = 2^{26}$ code vectors to be stored at receiver, which is 10^9 bits approximately.

If $YH^T = (0\,0\,0 \ldots 0)$, the $Y = X$ i.e. there is no error

But $XY \neq (0\,0\,0 \ldots 0)$, then $Y \neq X$ i.e. error exists in the received codeword.

For a code vector $X = (0111000)$ and the parity check matrix given below prove the $XH^T = (0, 0, 0 \ldots 0)$

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Solution:

$$H^T = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\therefore XH^T = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0+1+1+0+0+0+0 & 0+1+0+1+0+0+0 & 0+0+1+1+0+0+0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$$

$\therefore XH^T = (0, 0, 0)$

The received vector for the (7,4) code is $Y = 1001101$. Find the transmitted codeword using syndrome decoding technique. Given that -

$$H = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Solution:

$$H^T = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\therefore [S] = YH^T = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1+0+0+1+1+0+0 & 1+0+0+1+0+0+0 & 1+0+0+0+0+0+ \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$[1\ 0\ 0]$ is found in the 5th position of the matrix. So, error pattern is in, $E = 0000100$

We know, $X = Y + E$

$$\begin{array}{rl} Y = & 1001101 \\ \oplus\ E = & 0000100 \\ \hline X = & 1001001 \end{array}$$

Given that, $H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$, $X = (1001011)$, $Y = (1001001)$

Find the error pattern.

Solution:

Syndrome, $[S] = YH^T$

$$= [1001001] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$$= [1 \ 1 \ 1]$$

$[1\ 1\ 1]$ is found on the 6th position. So, error pattern is

$E = (0000010)$

If all the elements of $S = 0$, then $Y = X$ and there are no transmission errors. Errors are indicated by the presence of non-zero elements in S.

Standard Array Decoding:

Standard array decoding is a table look-up decoding technique for block codes. For an (n,k) binary block code C, there exist 2^n possible n-tuples that the decoder may receive, 2^k of which are the codewords. The standard array decoding method first arranges all $2^n - n$ tuples in an array as follows:

1. List all 2^k codewords among the 2^n n-tuples on the first row, starting with all-zero codeword 0
2. From the remaining n-tuples, choose one with the smallest number of 1's, and put it on the first position of the next row.
3. Add the chosen n-tuple to the codewords at the top of each column, and place the results in the corresponding positions of the row. The results are then removed from the remaining n-tuples.
4. Repeat steps 2 and 3 until all 2^n n-tuples have been used.

Consider a simple (5,2) code with the generator matrix $G = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$. The standard array of the code is constructed as follows- [00][G], [01][G], [10][G], [11][G]

00000	10101	01011	11110	← Codeword
10000	00101	11011	01110	
01000	11101	00011	10110	
00100	10001	01111	11010	
00010	10111	01001	11100	
00001	10100	01010	11111	
11000	01101	10011	00110	
10010	00110	11001	01100	

Coset Leader

Suppose the decoder receives $r = (01010)$ which contains an error in fifth position, the coset leader corresponding to r is (00001) and the codeword is (01011)

For a (6,3) linear block code -

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Solution:

$$S_0 = [000]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [000000]$$

$$S_1 = [001]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [001011]$$

$$S_2 = [010]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [010110]$$

$$S_3 = [011]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [011101]$$

$$S_4 = [100]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [100101]$$

$$S_5 = [101]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [101110]$$

$$S_6 = [110]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [110011]$$

$$S_7 = [111]\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [111000]$$

Standard array of the code is constructed below -

000000	001011	010110	100101	011101	101110	110011	111000
000001	001010	010111	100100	011100	101111	110010	111001
000010	001001	010100	100111	011111	101100	110001	111010
000100	001111	010010	100001	011001	101010	110111	111100
001000	000011	011110	101101	010101	100110	111011	110000
010000	011011	000110	110101	001101	111110	100011	101000
100000	101011	110110	000101	111101	001110	010011	011000
100010	$\boxed{101001}$	110100	000111	111111	001100	010001	011010

Notes: ${}^6C_2 = 15$. Total 15টি Hamming weight 2 codeword এর, single bit error ঠিকমত বাদি থাকে ${}^6C_2 - 12 = 3$টি Codeword]

If the decoder receives $r = (101001)$, which contains an error in eight position. In the above array we can find that the coset leader corresponding to r is (100010). So, the codeword is (001011).

To find out syndrome - $S = eH^T$

$$= [e_1\ e_2\ e_3\ e_4\ e_5\ e_6]\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= [e_1+e_2+e_4 \quad e_2+e_3+e_5 \quad e_1+e_3+e_6]$$

Error Pattern	Error Syndrome
000000	000
000001	001
000010	010
000100	100
001000	011
010000	110
100000	101
100010	111

Hamming Code: (7,4) Hamming Code

Hamming code is a error detecting + error correcting code.

For $n = m + k$, where n = length of the code

m = data bits

k = parity bits

$2^k \geq m + k + 1$

Parity bits at the position $2^0, 2^1, 2^2 \ldots$

D_4	D_3	D_2	P_3	D_1	P_2	P_1
7	6	5	4	3	2	1

Binary sequences -

P_3 P_2 P_1
000
1- 001
2- 010
3- 011
4- 100
5- 101
6- 110
7- 111

$P_1 = D_1 \oplus D_2 \oplus D_4$

$P_2 = D_1 \oplus D_3 \oplus D_4$

$P_3 = D_2 \oplus D_3 \oplus D_4$

If data bits are 1010, Find the codeword.

Solution:

1	0	1	P_3	0	P_2	P_1
7	6	5	4	3	2	1

$P_1 = 0 \oplus 1 \oplus 1 = 0$

$P_2 = 0 \oplus 0 \oplus 1 = 1$

$P_3 = 1 \oplus 0 \oplus 1 = 0$

$\therefore$ Transmitted bits = 1010010

Binary sequences, C calculation -

C_3 C_2 C_1
000
1- 001
2- 010
3- 011
4- 100
5- 101
6- 110
7- 111

$C_1 = D_1 \oplus D_2 \oplus D_4 \oplus P_1$ or, 1, 3, 5, 7

$C_2 = D_1 \oplus D_3 \oplus D_4 \oplus P_2$ or, 2, 3, 6, 7

$C_3 = D_2 \oplus D_3 \oplus D_4 \oplus P_3$ or, 4, 5, 6, 7

If received codeword is 1110010

1	1	1	0	0	1	0
D_4	D_3	D_2	P_3	D_1	P_2	P_1

$C_1 = 0 \oplus 1 \oplus 1 \oplus 0 = 0$, even

$C_2 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$, odd

$C_3 = 1 \oplus 1 \oplus 1 \oplus 0 = 1$, odd

C_3	C_2	C_1	
1	1	0	= 6

There is an error on 6 no bit.

When a code is received the following procedure is used to detect and correct the error:

1. Check even parity on positions 1,3,5,7. If it passes, then $C_1 = 0$ then it passes, else $C_1 = 1$
2. Check even parity on positions 2,3,6,7. If it passes, then $C_2 = 0$ else $C_2 = 1$
3. Check even parity on positions 4,5,6,7. If it passes, then $C_4 = 0$, else $C_4 = 1$
4. The decimal equivalent of $C_4C_2C_1$ gives the position of the incorrect bit and that bit is corrected. If $C_4C_2C_1 = 0$ then there is no error.

Iftekhar Sir:

Hamming codes are (n,k) codes with following properties-

1. $n = 2^q - 1$, q = no of parity bits
2. $k = 2^q - q - 1$, K = no of information bits
3. $q \geq 3$ i.e. minimum no of parity bits

Information bits, $I = (i_1, i_2, i_3, i_4)$, Parity bits, $P = (P_1, P_2, P_3)$

Parity bits:

$P_1 = i_1 + i_2 + i_3$, $P_2 = i_2 + i_3 + i_4$, $P_3 = i_1 + i_2 + i_4$

$C = (i_1, i_2, i_3, i_4, P_1, P_2, P_3)$

For $I = 0101$,

$P_1 = 0 \oplus 1 \oplus 0 = 1$

$P_2 = 1 \oplus 0 \oplus 1 = 0$

$P_3 = 0 \oplus 1 \oplus 1 = 0$

$\therefore C = \underline{0101100}$

(7,4) Hamming Code

0000 → 0000000

0001 → 0001011

0010 → 0010110

0011 → 0011101

⋮

1111 → 1111000

Decoding:

Received Word, $V = (v_1, v_2, v_3, v_4, v_5, v_6, v_7)$

For error checking, 3- parity checksum is used -

$S_1 = v_1 + v_2 + v_3 + v_5$

$S_2 = v_2 + v_3 + v_4 + v_6$

$S_3 = v_1 + v_2 + v_4 + v_7$

Error Syndrome $S = (S_1, S_2, S_3)$

Syndrome Table for (7,4) Hamming code, $H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$

Error Pattern	Error Syndrome
0000000	000 (0)
0000001	001 (1)
0000010	010 (2)
0000100	100 (4)
0001000	011 (3)
0010000	111 (7)
0100000	110 (6)
1000000	101 (5)

If received bit is $V = 1011001$, Find out the transmitted bit.

Solution: Given, $V = 1011001$

$S_1 = 1+0+1+0 = 0$

$S_2 = 0+1+1+0 = 0$

$S_3 = 1+0+1+1 = 1$

$\therefore$ Syndrome, $S = (001)$; Error Pattern, $e = 0000001$

$\therefore$ Transmitted bit, $c = v + e$

$= 1011001 + 0000001$

$= 1011000$