WEEK 8

DISCRETE MEMORYLESS CHANNELS

Discrete memory less channel, the counterpart of a discrete memoryless source is a statistical model with an input X and an output Y that is a noisy version of X; both X and Y are random variables. Every unit of time, the channel accepts an input symbol X selected from an alphabet x and, in response, it emits an output symbol Y from an alphabet y . The channel is said to be "discrete" when both of the alphabets x and y have $finite$ sizes. It is said to be "memoryless" when the current output symbol depends $only$ on the current input symbol and not any of the previous ones.

The channal is decribed in term of the The input alphabet

$x=\{x_0 , x_1 , \ldots , x_{j-1}\}$

& the output alphabrt

$y=\{y_0 , y_1 , \ldots , y_{k-1}\}$

& a set of transition probability

$$p(y_k|x_j) = P(Y = y_k \,|X = x_j)$$

Given that we have $0 \leq p(y_k|x_j) \leq 1$

A convenient way of describing a discrete memoryless channel is to arrange the various transition probabilities of the channel in the form of a matrix as follows:

A convenient way of describing a discrete memoryless channel is to arrange the various transition probabilities of the channel in the form of a matrix as follows:

$$\mathbf{P} = \begin{bmatrix} p(y_0|x_0) & p(y_1|x_0) & \cdots & p(y_{K-1}|x_0) \\ p(y_0|x_1) & p(y_1|x_1) & \cdots & p(y_{K-1}|x_1) \\ \vdots & \vdots & & \vdots \\ p(y_0|x_{J-1}) & p(y_1|x_{J-1}) & \cdots & p(y_{K-1}|x_{J-1}) \end{bmatrix}$$

The *J-by-K* matrix P is called the *channel matrix*, or *transition matrix.* Note that each *row* of the channel matrix P corresponds to a *fixed channel input*, whereas each column of the matrix corresponds to a *fixed channel output.* Note also that a fundamental property of the channel matrix P, as defined here, is that the sum of the elements along any row of the matrix is always equal to one; that is,

$$\sum_{k=0}^{K-1} p(y_k|x_j) = 1$$

Having specified the random variable X denoting the channel input, we may now specify the second random variable *Y* denoting the channel output. The *joint probability distribution* of the random variables *X* and Y is given by:

$$
\begin{aligned}
P(x_j, y_k) &= P(X = x_j, Y = y_k) \\
&= P(Y = y_k \mid X = x_j)P(X = x_j) \\
&= P(y_k \mid x_j)p(x_j)
\end{aligned}
$$

The *marginal probability distribution* of the output random variable *Y* is obtained by averaging out the dependence of $p(x_j, y_k)$ on x_j , as shown by:

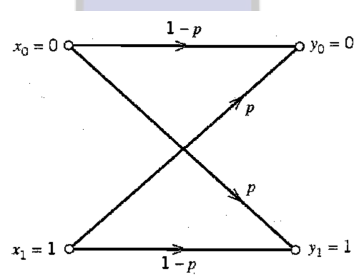
$$p(y_k) = P(Y = y_k)$$

$$= \sum_{j=0}^{j-1} P(Y = y_k | X = x_j)P(X = x_j) = \sum_{j=0}^{j-1} p(y_k|x_j)p(x_j)$$

The probabilities $p(x_j)$ are known as the *a priori probabilities* of the various input symbols. Equation above states that if we are given the input *a priori* probabilities $p(x_j)$ and the channel matrix [i.e., the matrix of transition probabilities $P(y_k \mid x_j)$] then we may calculate the probabilities of the various output symbols, the $p(y_k)$.

Binary Symmetric Channel

The *binary symmetric channel* is of great theoretical interest and practical importance. It is a special case of the discrete memoryless channel with $J = K = 2$. The channel has two input symbols ($x_0 = 0$, $x_1 = 1$) and two output symbols ($y_0 = 0$, $y_1 = 1$). The channel is symmetric because the probability of receiving a 1 if a 0 is sent is the same as the probability of receiving a 0 if a 1 is sent. This conditional probability of error is denoted by *p.* The *transition probability diagram* of a binary symmetric channel is as shown in Figure below:



$$P_{10} = P(y = 1 / x = 0)$$
$$P_{01} = P(y = 0 / x = 1)$$
$$p_{10} = p_{01} = p$$

ERROR-FREE COMMUNICATION OVER A NOISY CHANNEL

As seen previously, messages of a source with entropy $H($m) can be encoded by using an average of H(m) digits per message. This encoding has zero redundancy. Hence, if we transmit these coded messages over a noisy channel, some of the information will be received erroneously. There is absolutely no possibility of error-free communication over a noisy channel when messages are encoded with zero redundancy. The use of redundancy, in general, helps combat noise. This can be seen from a simple example of a **single parity-check code,** in which an extra binary digit is added to each code word to ensure that the total number of I s in the resulting code word is always even (or odd). If a single error occurs in the received code word, the parity is violated, and the receiver requests retransmission. This is a rather simple example to demonstrate the utility of redundancy.

The addition of an extra digit increases the average word length to H(m) + 1, giving $\boldsymbol{\eta} = H(m)/[H(m) + 1]$, and the redundancy is 1 - $\boldsymbol{\eta} = 1/[H(\text{m}) + 1]$. Thus, the addition of an extra check digit increases redundancy, but it also helps combat noise. Immunity against channel noise can be increased by increasing the redundancy. Shannon has shown that it is possible to achieve error-free communication by adding sufficient redundancy. For example, if we have a **binary symmetric channel (BSC)** with an error probability P_e, then for error- free communication over this channel, messages from a source with entropy H (m) must be encoded by binary codes with a word length of at least $H(m)/C_s$, where

$$C_s = 1 - [P_e \log\frac{1}{P_e} + (1 - P_e)\log\frac{1}{1 - P_e}]$$

The parameter C_s (C_s < 1) is called the **channel capacity.**

The efficiency of these codes is never greater than C_s. If a certain binary channel has C_s = 0.4, a code that can achieve error-free communication must have at least 2.5 $H(m)$ binary digits per message, which is two-and-one-half times as many digits as required for coding without redundancy. This means there are 1.5 $H($m) redundant digits per message. Thus, on the average, for every 2.5 digits transmitted, one digit is the information digit and 1.5 digits are redundant, or check digits, giving a redundancy of 1 - C_s = 0.6.

As discussed earlier, P_e, the error probability of binary signaling, varies as e^{-kEb} and, hence, to make $P_e \rightarrow 0$, either $S_i \rightarrow \infty$ or $R_b \rightarrow 0$. Because S_i must be finite, $P_e \rightarrow 0$ only if $Rb \rightarrow 0$. But Shannon's results state that it is really not necessary to let $R_b \rightarrow 0$ for error-free communication. All that is required is to hold R_b below C, the channel capacity per second.