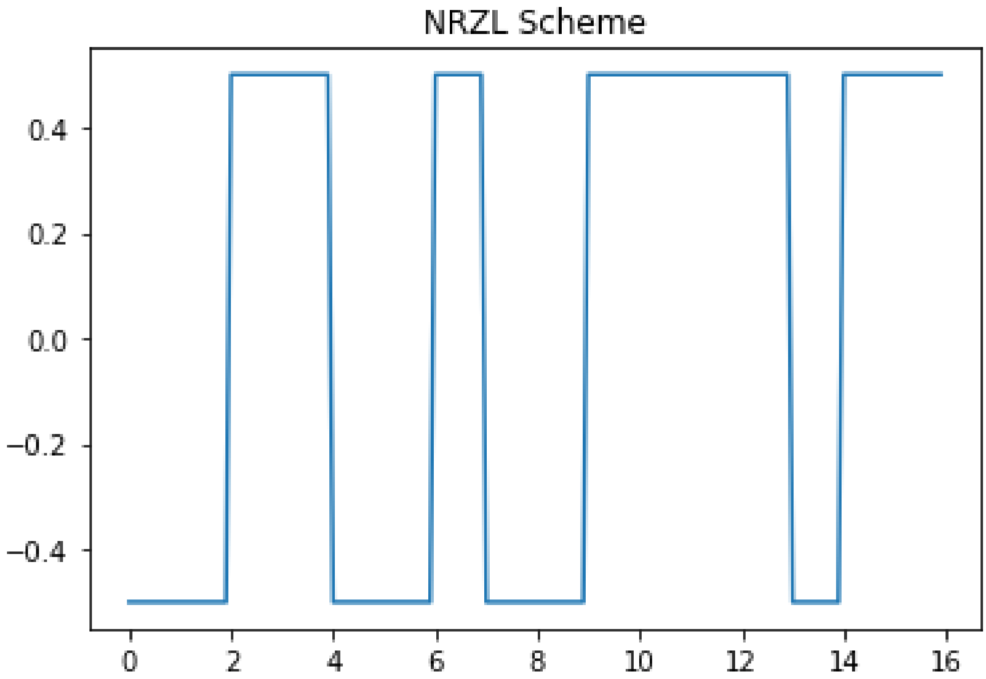
1. First, we generate a random n-bit stream corresponding to a Binary PCM NRZ-L waveform. Let it be represented by x(t). A binary PCM signaling scheme can be defined as $s_1(t)$ for bit 1 and $s_2(t)$ for bit 0. Here,

$$\begin{aligned} s_1(t) &= +A, \ \ 0 \leq t \leq T \\ &= 0, \quad \text{elsewhere} \\ s_2(t) &= A, \ \ 0 \leq t \leq T \\ &= 0, \quad \text{elsewhere} \end{aligned}$$

The waveform is shown below:



Next, we sample the signal with sampling frequency $f_s = \frac{10}{T}$, or 10 times the bit frequency. Thus we get 10 samples per bit, giving a total of $10 \cdot n$ samples.

Next we generate AWGN using the randn() function of length $10 \cdot n$. This will give us samples with mean zero and variance 1. In order to get the desired variance we multiply these noise samples with the standard deviation.

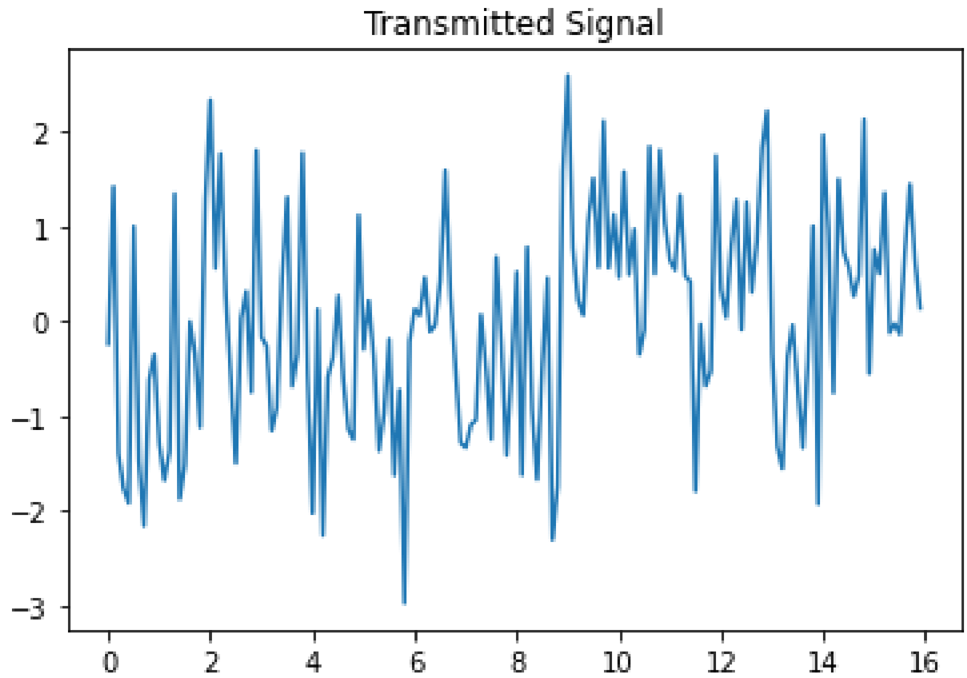
Now we declare a matched filter. A matched filter is obtained by correlating a known delayed signal, or template, with an unknown signal to detect the presence of the template in the unknown signal. This is equivalent to convolving the unknown signal with a conjugated time-reversed version of the template. Impulse response of this filter is given by

$$\begin{aligned} h(t) &= s_1(T-t) - s_2(T-t) \\ &= 2A \end{aligned}$$

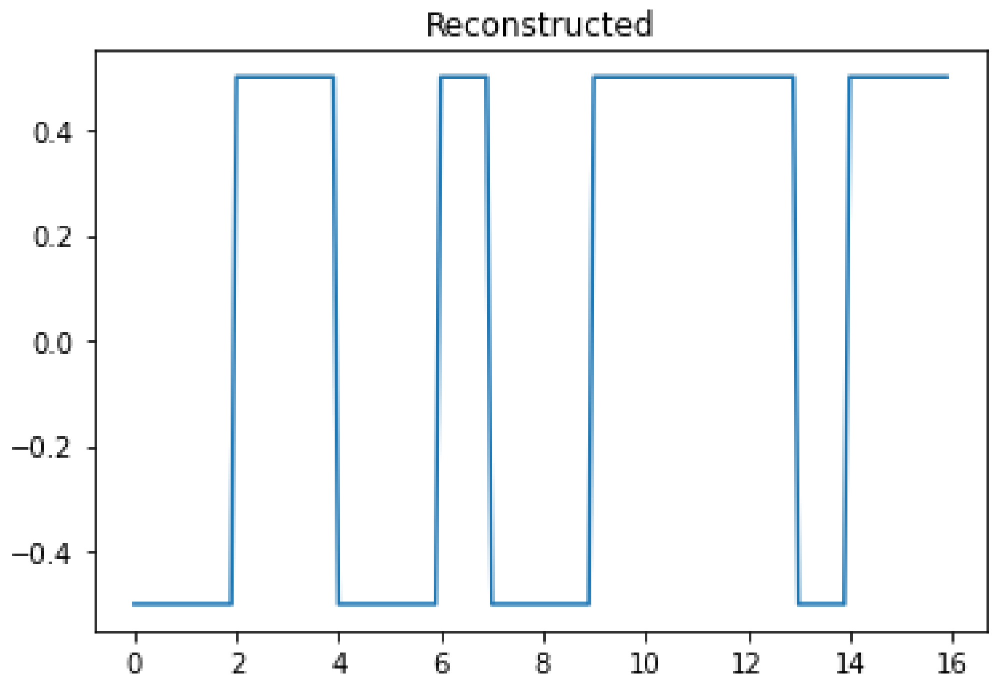
We need to find the threshold for this filter, which in our case is given by

$$h(t) * s_1(t) + h(t) * s_2(t) = 0$$

We add the noise to the signal to get r(t) i.e. the distorted signal.
The transmitted noisy signal is given below.

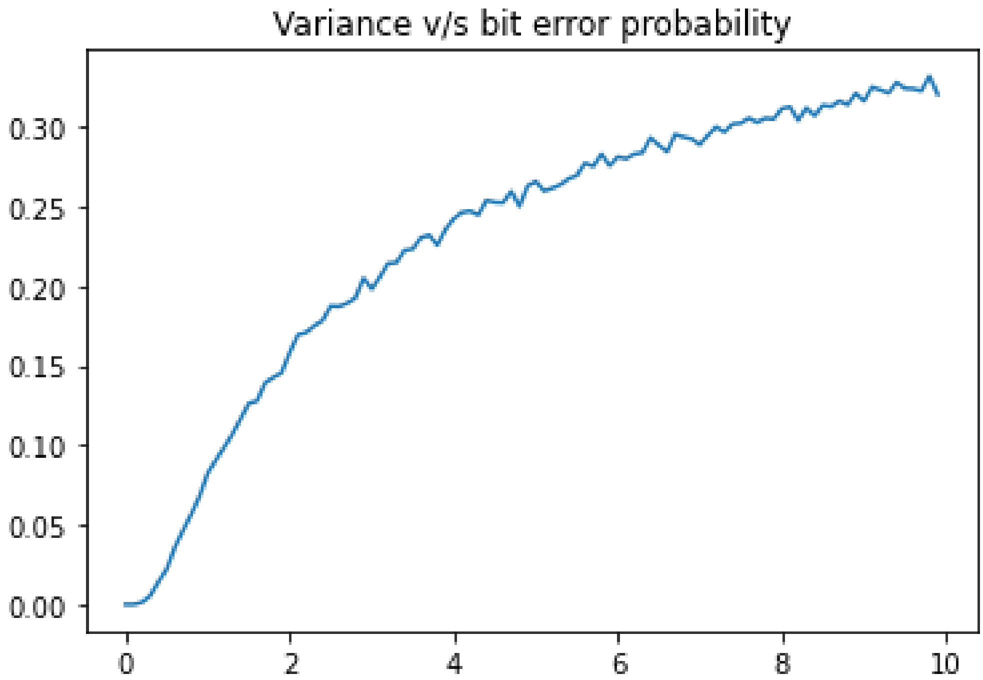


Now we need to pass the samples of this signal bit by bit through this filter. The output is given by the dot product of h(t) and the 10 bit samples for all n bits. For each of these n outputs, we sum the 10 values of the samples and compare it with threshold value. Predicted output is bit 1 if this sum is greater than 0 and bit 0 if the sum is less than 0. The reconstructed signal using this algorithm is given below.

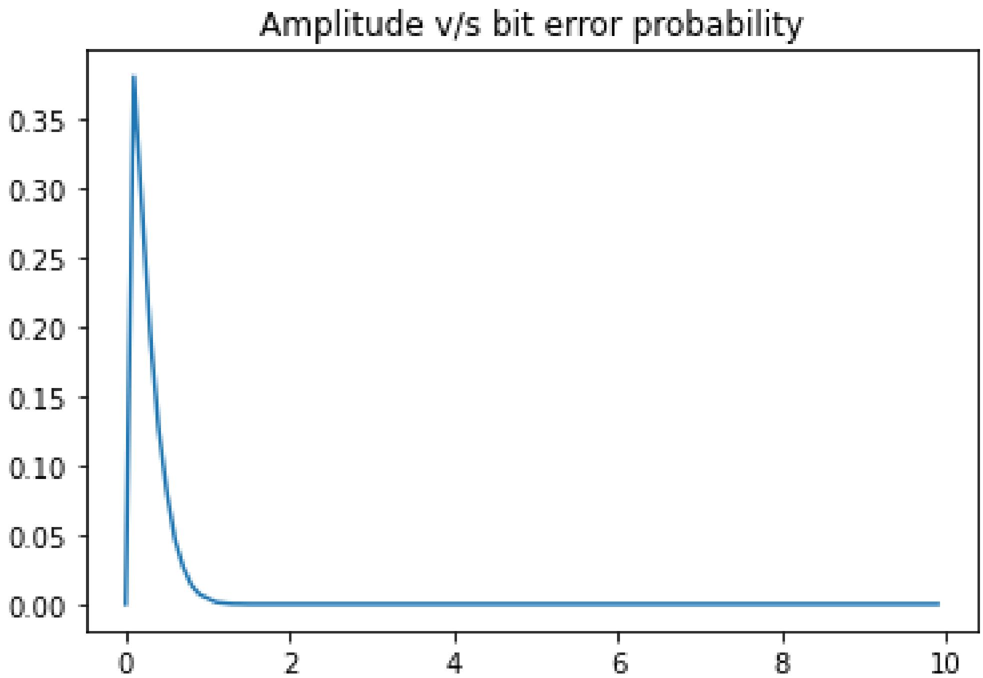


We repeat this experiment around 1000 times for different values of variance and plot the average value of relative error by comparing the output with input and plotting the error percentage graph w.r.t the variance values of noise signal.

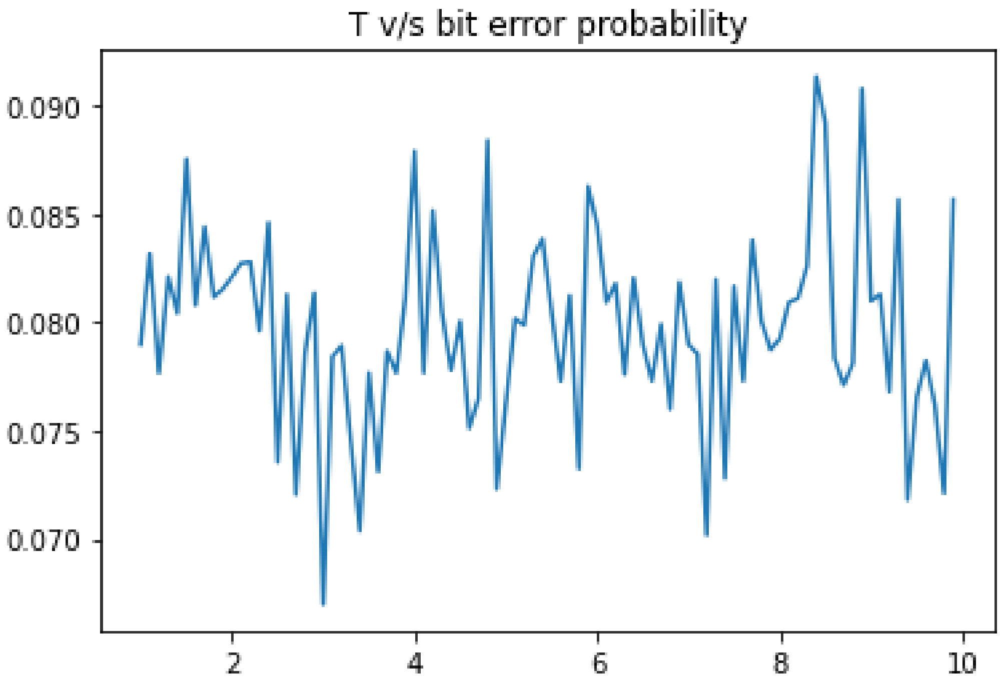
Observations:



1. As the noise variance increase the error rate increases non-linearly.



2. Now we increase the amplitude A x(t) and perform the same experiment with same values of T. We notice that the relative error decreases. This is because the SNR ratio increases and hence the performance of matched filter increases.



3. Finally, we increase T keeping the value of A same as that in the first plot.