Mohamed Nagy
20012409
LAB1 – BER

Code Snips

First generate random data and save it in data vector, then calculate the signal power which is the Expectations of $E(x^2)$, then generate SNR range in decibel (dB) and in linear scale, then create two vectors of length N to hold the received sequence which will be the original data + noise, then the received bits is the sampled bits after adding the noise.

```
clc; clear;

% Number of bits
N=1e6;

% Generate random binary data vector (sent signal)
data = randi([0,1],1,N);

% Calculate transmitted signal power =E(x^2)
sigPower = mean(data.^2);

% SNR range in dB
SNRdB_range = -20:2:20;

% Convert SNR to linear scale
SNR=db2pow(SNRdB_range);

% Initialize error count
errorNUM=0;

RecievedSequence = zeros(size(data));
RecievedBits = zeros(size(data));
BER = zeros(size(SNRdB_range));
```

Here we loop on the hole data sequence with the length of the SNR range, first we get the signal noise from the relation $noise = \frac{1}{\sqrt{SNR}} * randn$ then it will be divided by the sqrt power since it's not normalized, then we will add the noise to the original data, then we will loop to all of the data sequence to check the bit value with the specified threshold to reconstruct the signal, then we will XOR the original data sequence with the reconstructed sequence to check the changed bits as the XOR will give in case of the flipped bits then we will count the flipped bits to count them up to calculate the BER.

```
for k=1:length(SNRdB_range)
    % Add noise based on SNR
    % Dividing by the sqrt power since its not normalized
    noise=((1/sqrt(SNR(k)))*randn(1,N)*sqrt(sigPower));

    % Received signal with noise
    RecievedSequence = data+noise;

    for i=1:N
        %define the threeshold value
        if RecievedSequence(i)<0
            RecievedBits(i) = 0;
        else
            RecievedBits(i) = 1;
        end
    end

    %comapre original bit with the recived one
    Rx=xor(data,RecievedBits);

    % Count Number of Errored bits
    for j=1:N
        if(Rx(j)==1)
            errorNUM = errorNUM+1;
        end
    end

    BER(k) = errorNUM ./ N;
    errorNUM=0;
end
```

Then finally we will display the BER graph and the signal power.

```matlab
% Plot BER curve
figure;
semilogy(SNRdB_range, BER, 'x-k', 'Color', 'b', 'LineWidth', 1);
xlabel('SNR (dB)'); ylabel('Bit Error Rate (BER)');
title('Bit Error Rate vs. SNR'); grid on;
yticks(0:0.05:1);

% Calculation of transmitted signal power
disp(['Transmitted Signal Power: ' num2str(sigPower)]);

% Find SNR where the system is nearly without error
[min_BER, min_BER_index] = min(BER);
SNR_nearly_without_error = SNRdB_range(min_BER_index);
disp(['SNR where the system is nearly without error: ' num2str(SNR_nearly_without_error) ' dB']);
```

Command Window

```
Transmitted Signal Power: 0.49994
SNR where the system is nearly without error: 16 dB
>>
```