Harish Boggarapu

ECE 3522: Stochastic Processes in Signals and Systems

Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA 1912

# Problem Statement

In this assignment we are asked to create a sinewave signal with Gaussian white noise. We have to compute to compute the power of the signal as well as the autocorrelation and magnitude spectrum for values of signal to noise ratio. We have to observe the plots of the autocorrelation for different SNR’s. We also have to process a signal at SNR 30dB through a digital filter and compute autocorrelation and magnitude of the signal.

1. Approach and Results

Power of the signal with different SNR’s

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SNR (dB) | -30 | -20 | -10 | 0 | 10 | 20 | 30 |
| Power (W) | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.2 | 0.3 |

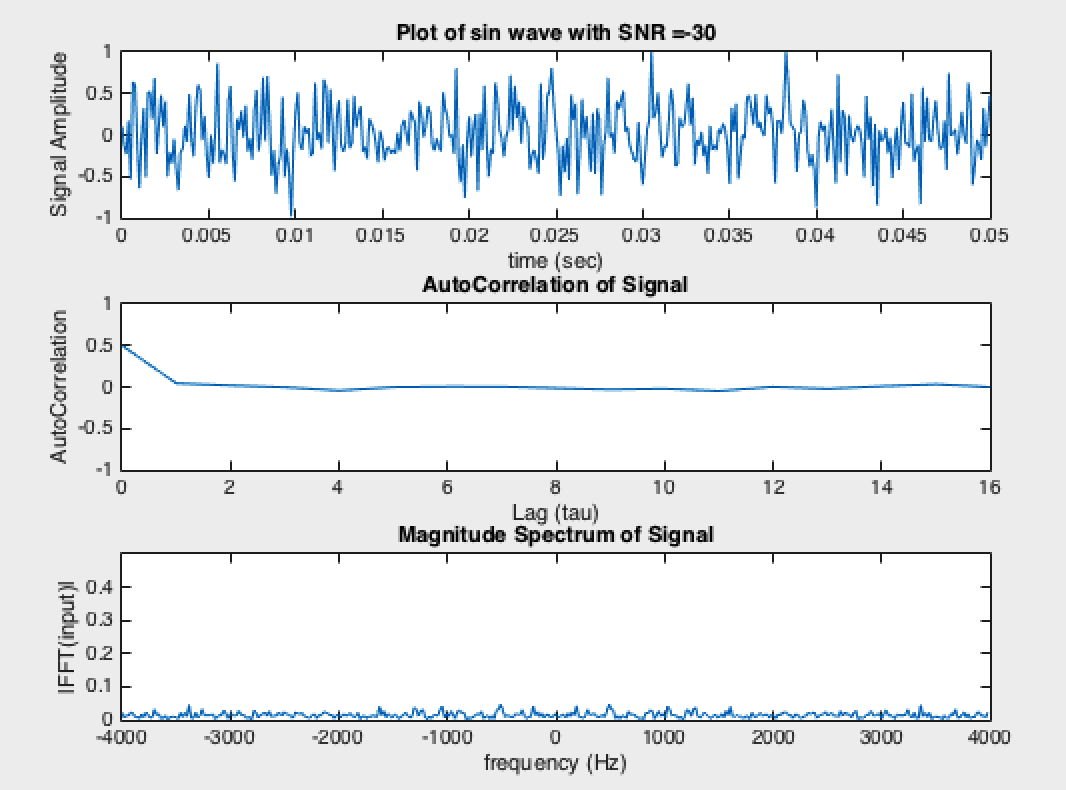
****

Figure : Plot of signal, autocorrelation and magnitude of SNR = -30

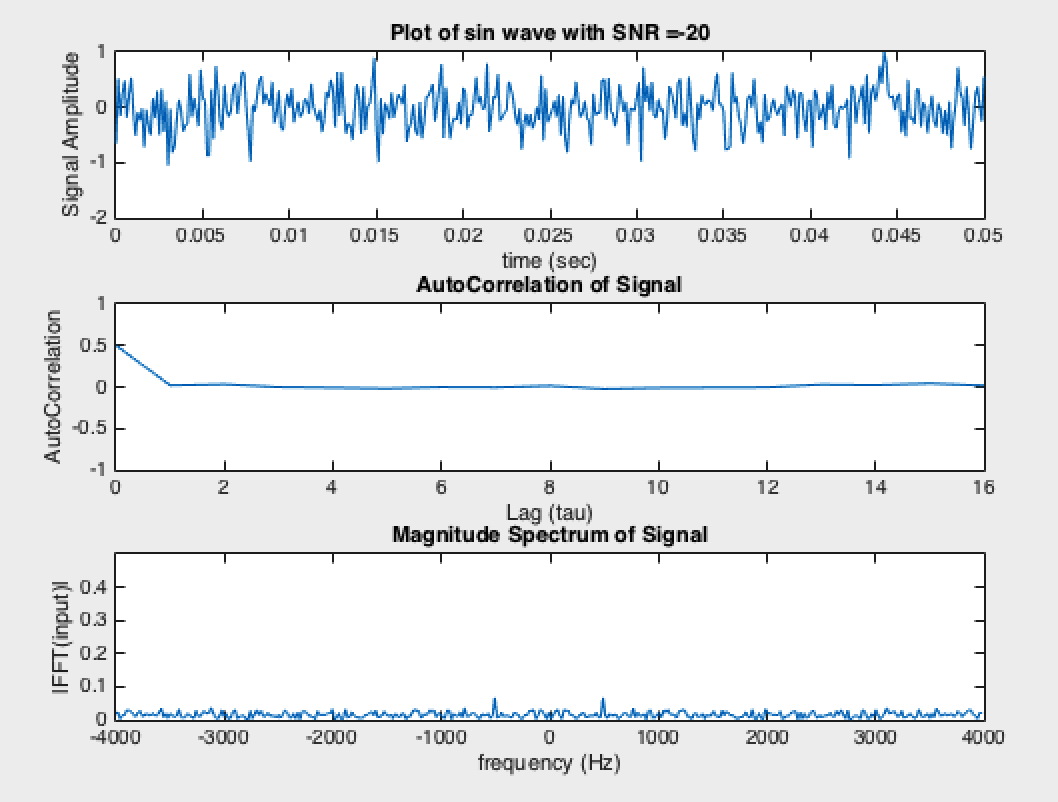
****

Figure : Plot of signal, autocorrelation and magnitude of SNR = -20

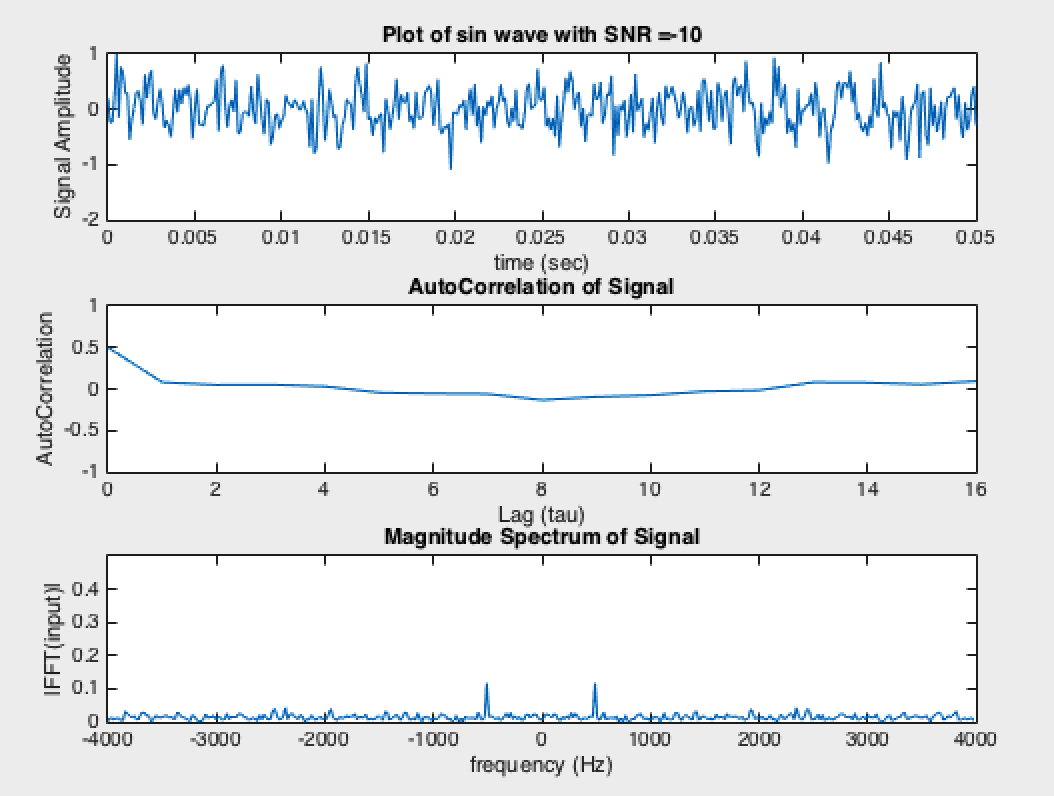
****

Figure : Plot of signal, autocorrelation and magnitude of SNR = -10

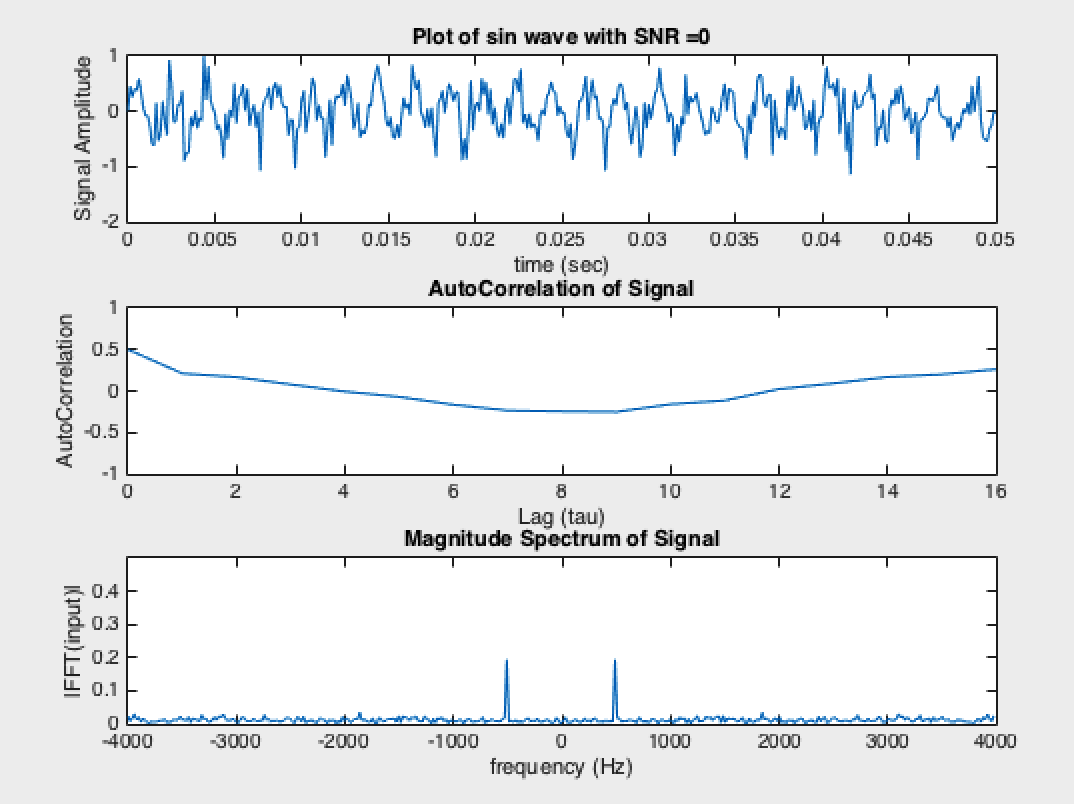
****

Figure : Plot of signal, autocorrelation and magnitude of SNR = 0

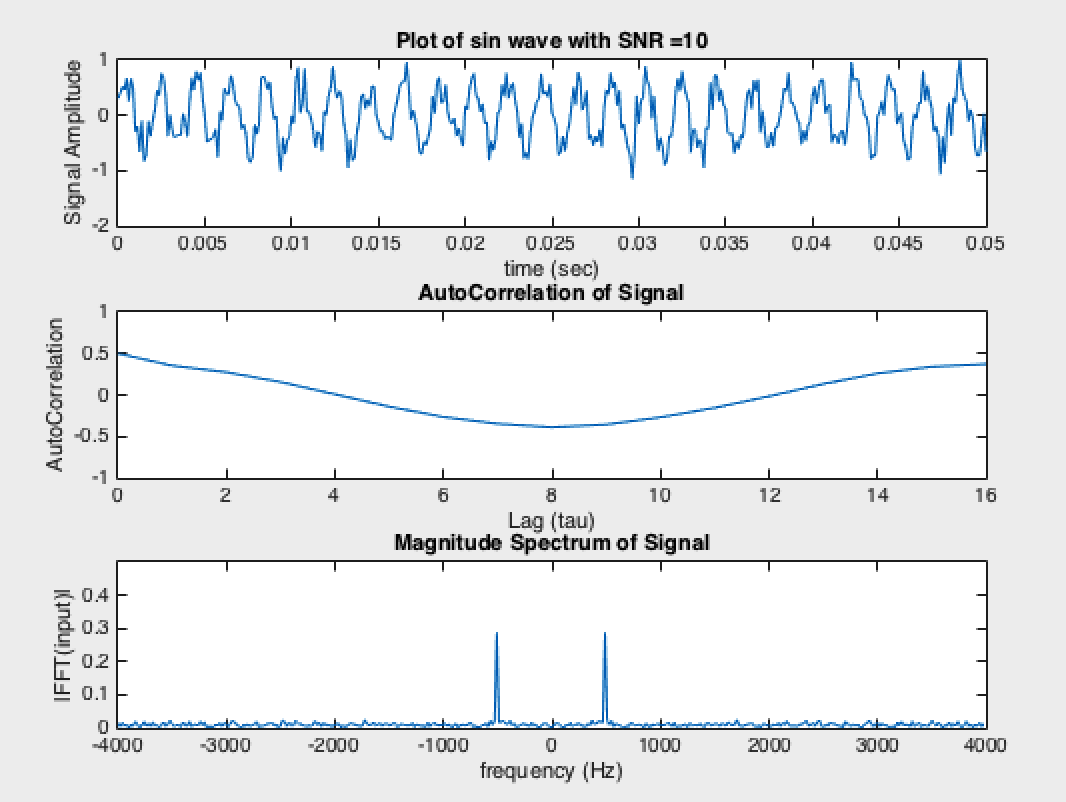
****

Figure : Plot of signal, autocorrelation and magnitude of SNR = 10

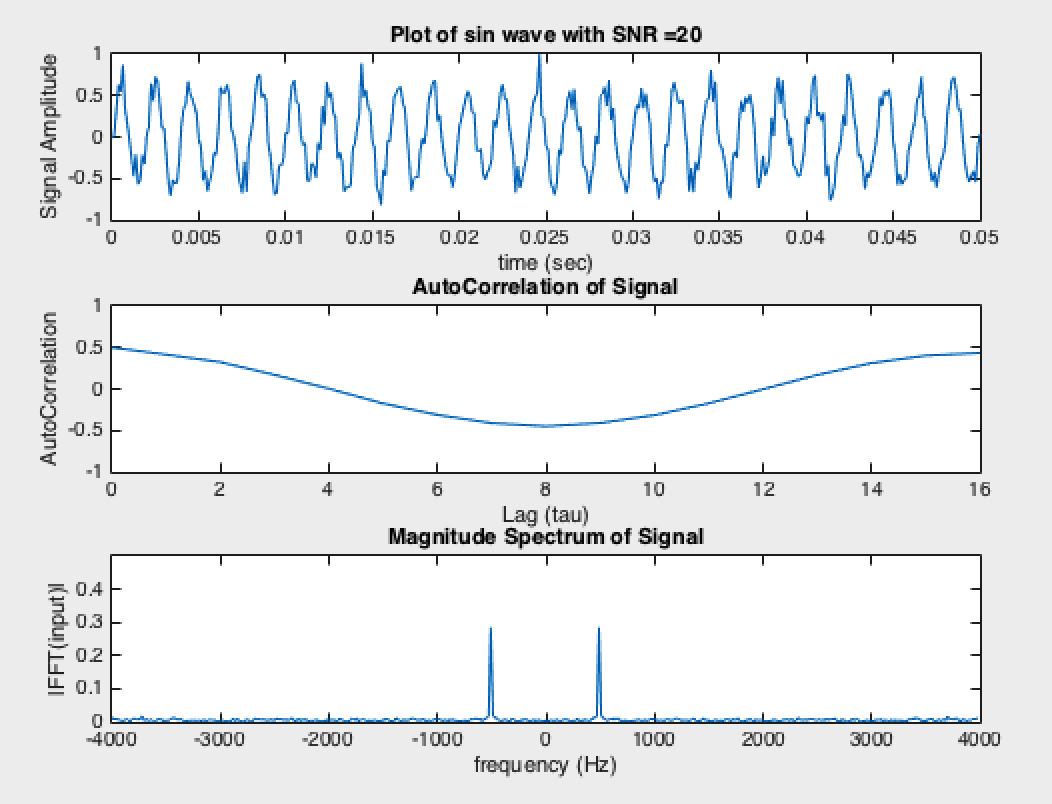
****

Figure : Plot of signal, autocorrelation and magnitude of SNR = 20

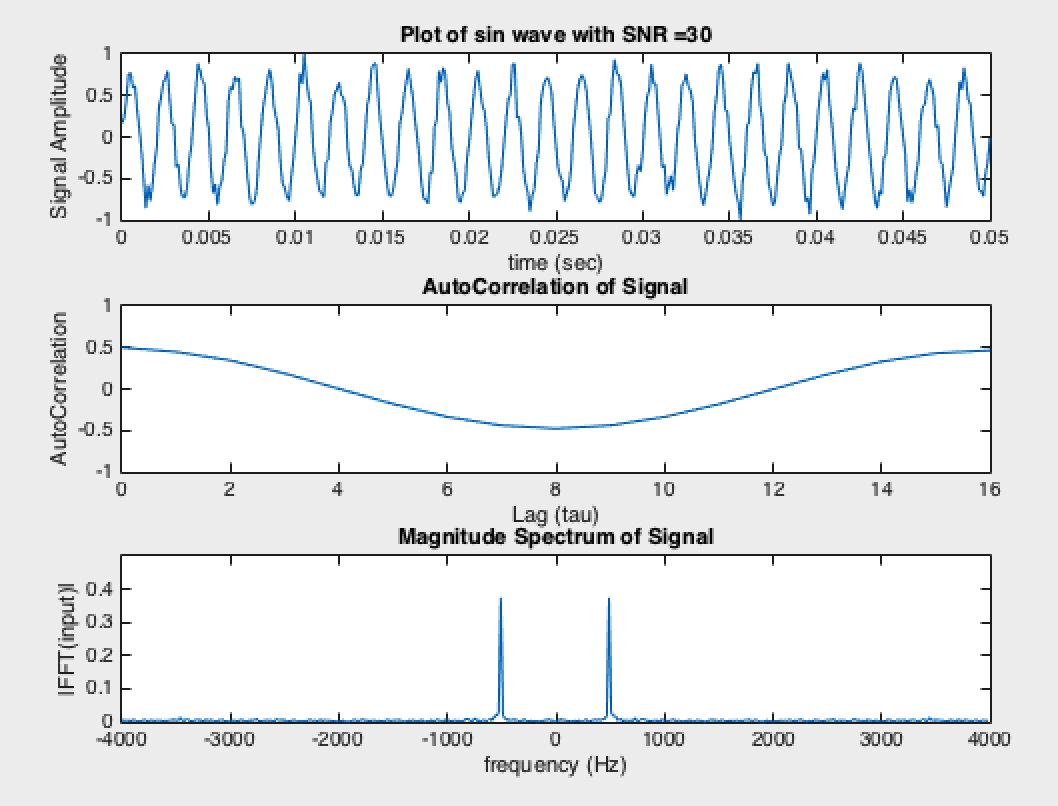
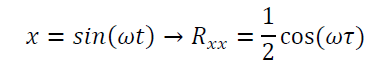
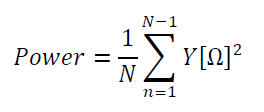


Figure : Plot of signal, autocorrelation and magnitude of SNR = 30

For a pure sine signal, the autocorrelation is computed using the formula



After we apply the filter, we find power using



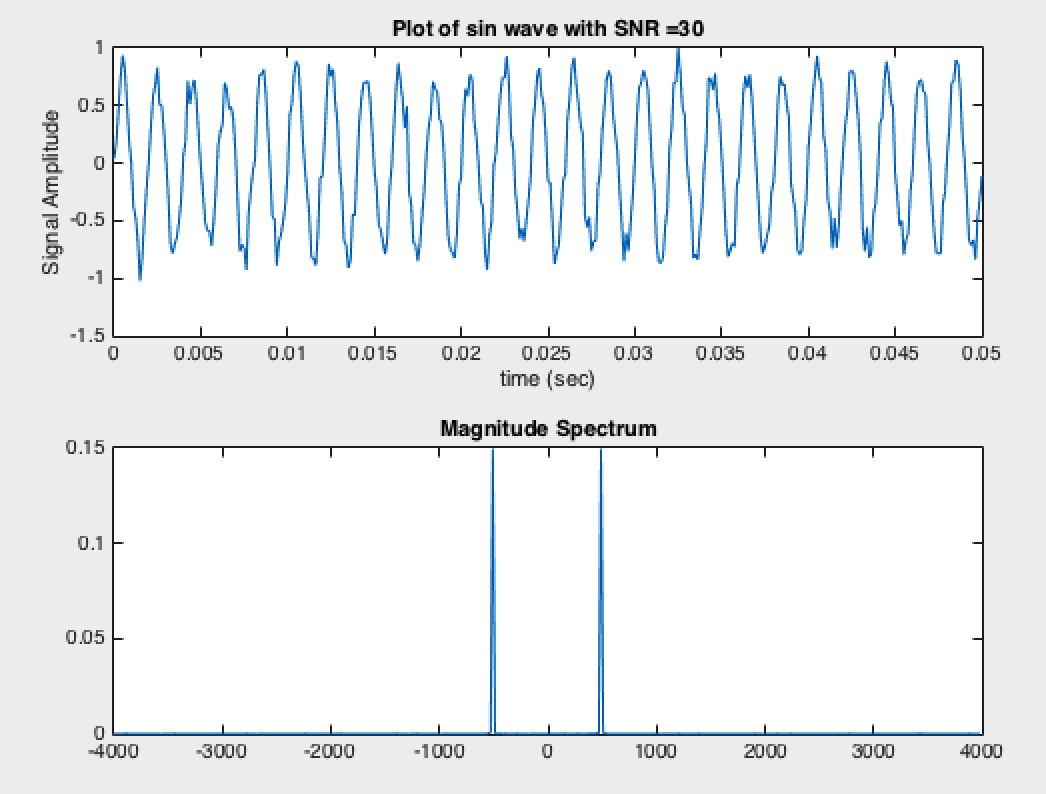


Figure : Plot of the signal and magnitude spectrum after filtering

1. MATLAB Code

clc; clear; close all;

t = 0:1/8000:0.05;

for SNR = -30:10:30

x = generate\_sine(1,500,.05,8000,SNR);

figure('name','Assignment 9');

subplot(3,1,1);

plot(t,x)

title(sprintf('Plot of sin wave with SNR =%d',SNR));

xlabel('time (sec)');

ylabel('Signal Amplitude');

subplot(3,1,2);

% Find AutoCorrelation for the first 16 lags

AutoCorr(x,16);

% Part 4

subplot(3,1,3);

% Fourier Tranform

FFTMag2(x, 8000);

title('Magnitude Spectrum of Signal');

ylim([0 0.5]);

end

for SNR = 30

% Generate the signal

x = generate\_sine(1,500,.05,8000,SNR);

subplot(2,1,1);

plot(t,x)

title(sprintf('Plot of sin wave with SNR =%d',SNR));

xlabel('time (sec)');

ylabel('Signal Amplitude');

% Insatiate the filtered signal

y(1) = x(1);

% Digital Filter

for n = 2:length(t)

y(n) = 0.5\*y(n-1)+x(n);

end

subplot(2,1,2);

acf = AutoCorr(y, length(t)-1);

ftACF = FFTMag2(acf\*2,8000);

df = 8000/length(t);

NFFT = 2^nextpow2(length(t));

f = (-8000/2) : df : (8000/2)-df;

FTY = fft(y)/length(t);

FTMY = abs(FTY);

plot(f,fftshift(FTMY, 2).^2);

FTX = fft(x)/length(t);

FTMX = abs(FTX);

plot(f,fftshift(FTMX, 2).^2);

title('Magnitude Spectrum');

end

function [ sig ] = generate\_sine(Amplitude,f, T, fs,snr\_db )

% time scale

t = 0:1/fs:T;

% num of samples

Nsamples = length(t);

% sine wave without noise

x = Amplitude\*sin(2\*pi\*f\*t);

% signal power

Sp = (Amplitude^2)/2;

% noise power

Np = Sp/db2mag(snr\_db);

% mean, variance and sigma

Nm = 0;

Nv = Np;

Nstd = sqrt(Nv);

NoiseSignal = myNormDist(Nm, Nstd, Nsamples);

% Create Noisy Signal and Normalize

sigNoisy = x + NoiseSignal;

sig = sigNoisy/max(sigNoisy)\*Amplitude;

end

function out = myNormDist(mu, sigma, N)

% Find two uniform Variables

x1 = rand(1,N);

x2 = rand(1,N);

y1 = sqrt(-2.\*log(x1)).\*cos(2.\*pi.\*x2);

out = (y1\*sigma + mu);

end

function FTM = FFTMag2(Signal, fs)

samL = length(Signal);

df = fs/samL;

NFFT = 2^nextpow2(samL);

f = (-fs/2) : df : (fs/2)-df;

% Take the fourier transform. The fftshift will duplicate our FT for

% negative frequencies. (Plot is centered at zero)

FT = fft(Signal)/samL;

FTM = abs(FT);

plot(f,fftshift(FTM, 2));

xlabel('frequency (Hz)')

ylabel('|FFT(input)|');

title('Magnitude Spectrum');

end

function acf1 = AutoCorr(Signal, lags)

[acf,lags,bounds] = autocorr(Signal,lags);

acf = acf\*max(Signal)^2/2;

plot(lags, acf);

maxA = round(max(acf));

ylim([-maxA maxA]);

xlabel('Lag (tau)');

ylabel('AutoCorrelation');

title('AutoCorrelation of Signal');

acf1 = acf;

end

# Conclusions

This assignment enforces the relationship between SNR and autocorrelation and magnitude spectrum. SNR is defined as the ratio of signal power to noise power. At SNR 0 dB, the amplitude of the signal and the noise fluctuations are similar. As you see in the first task, the peaks of the magnitude spectrum increases with the increase in the signal-to noise ratio. Also, the plots starts to dip with the increasing SNR’s. With higher SNR, the autocorrelation of the signal decreases negatively. So, at SNR = 30, the signal lower negative correlation compared to that at SNR less than 30. At SNR = 0, there is no correlation because plot of autocorrelation is horizontal at 0 for tau grater 1.