Tyler Castelli

ECE 3512: Signals – Continuous and Discrete

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

# Problem Statement

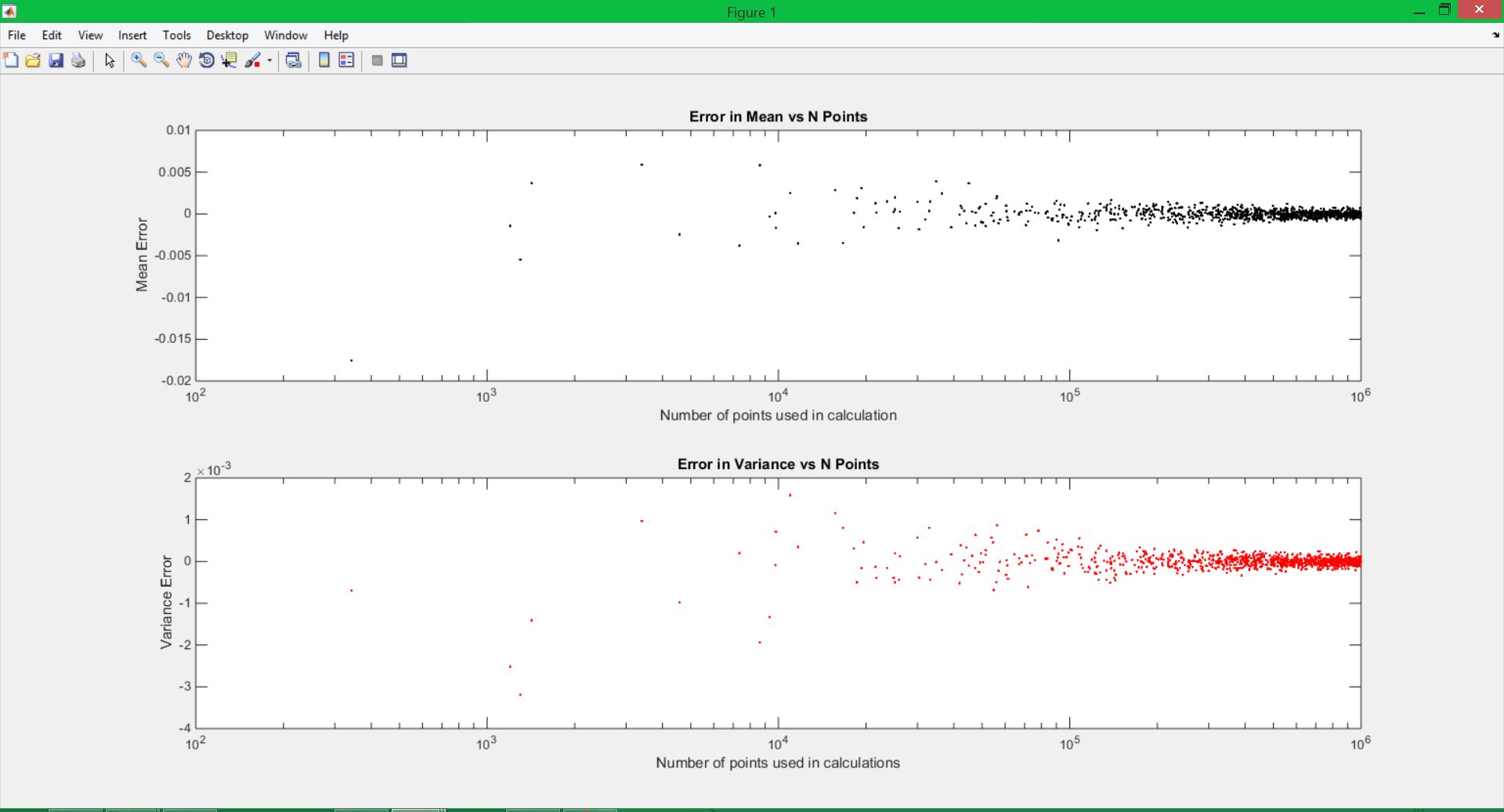
The objective of this assignment find the error between the measured values of a uniform distribution and the expected values We calculated the error in the mean, the variance, and the mean-squared error of pdf’s that all varied in number of samples ranging from [1,10^6].

# Approach and Results

For this assignment I wrote to functions to perform the calculations and return the results. The results were then stored in vectors and plotted. Figure 1 shows the error between the expected values and the measured values in both the mean and variance. The expected value for the mean is 0.5, and for the variance is 0.0833. These values can be calculated using from the mean and variance equations of a uniform distribution. The mean and variance can be found by:

The errors were found by subtracting the expected value from the measured values. The results are shown in the plots below.

Figure 1



The next three figures show the pdf’s of a three random signals that are distributed uniformly. Figure 2 is the pdf of a signal that has 10 points, figure 3 is the pdf of a signal that has 1,000 points, and figure 5 shows the pdf of a signal that has 1,000,000 points. Each pdf is represented through a histogram and utilizes a bin size of 0.01 to generate 100 bins.

Figure 2

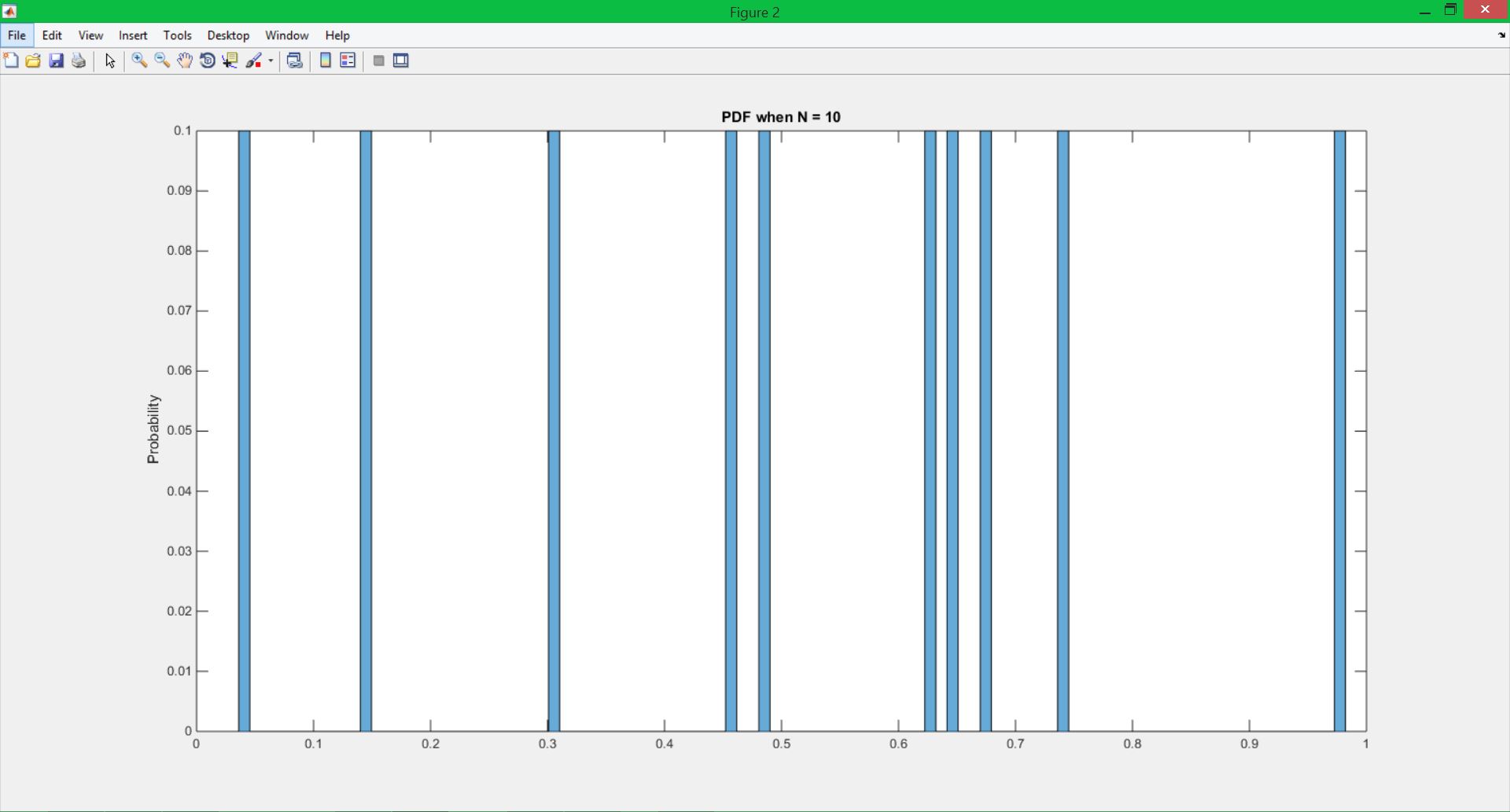


Figure 3

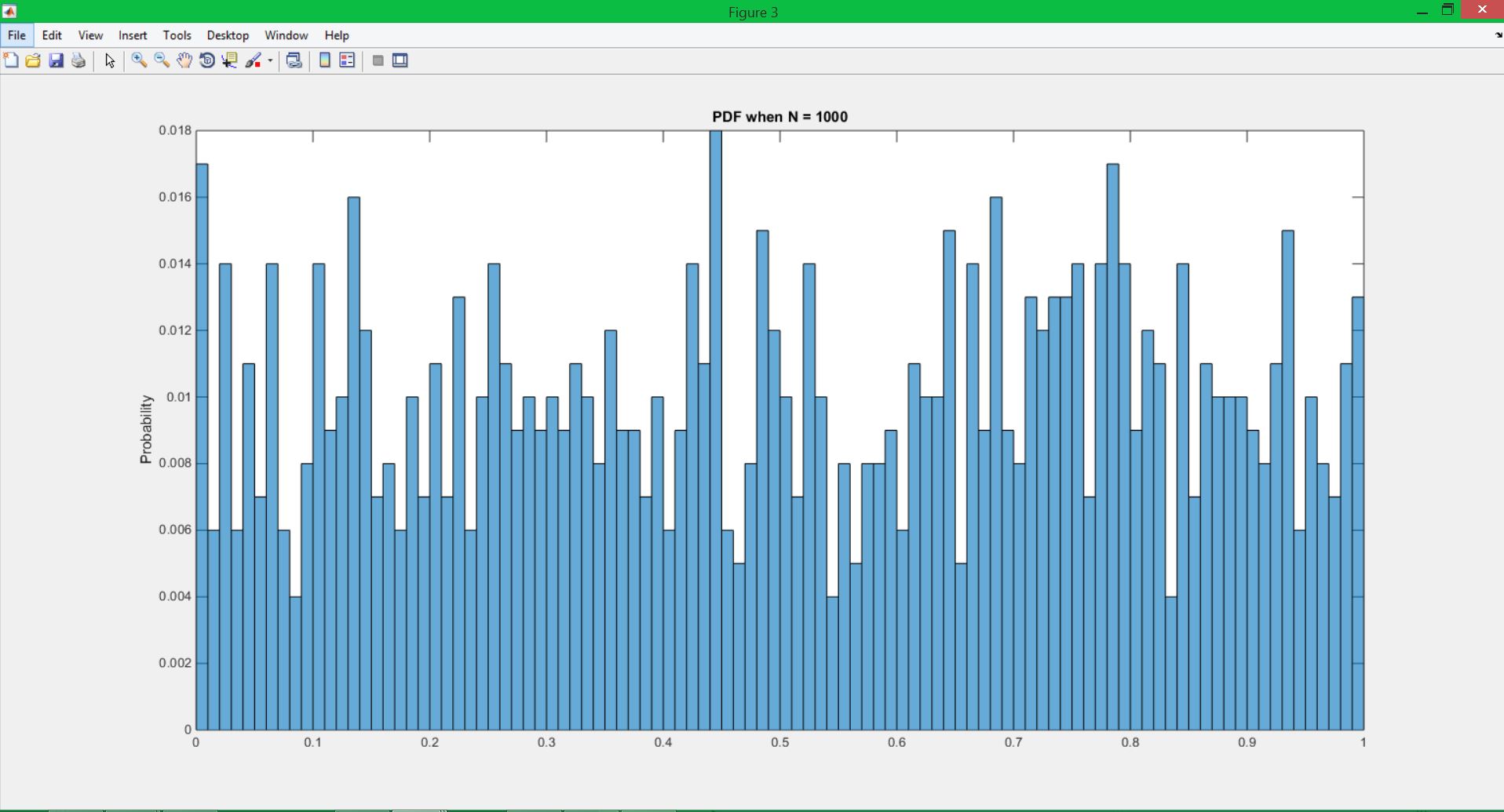
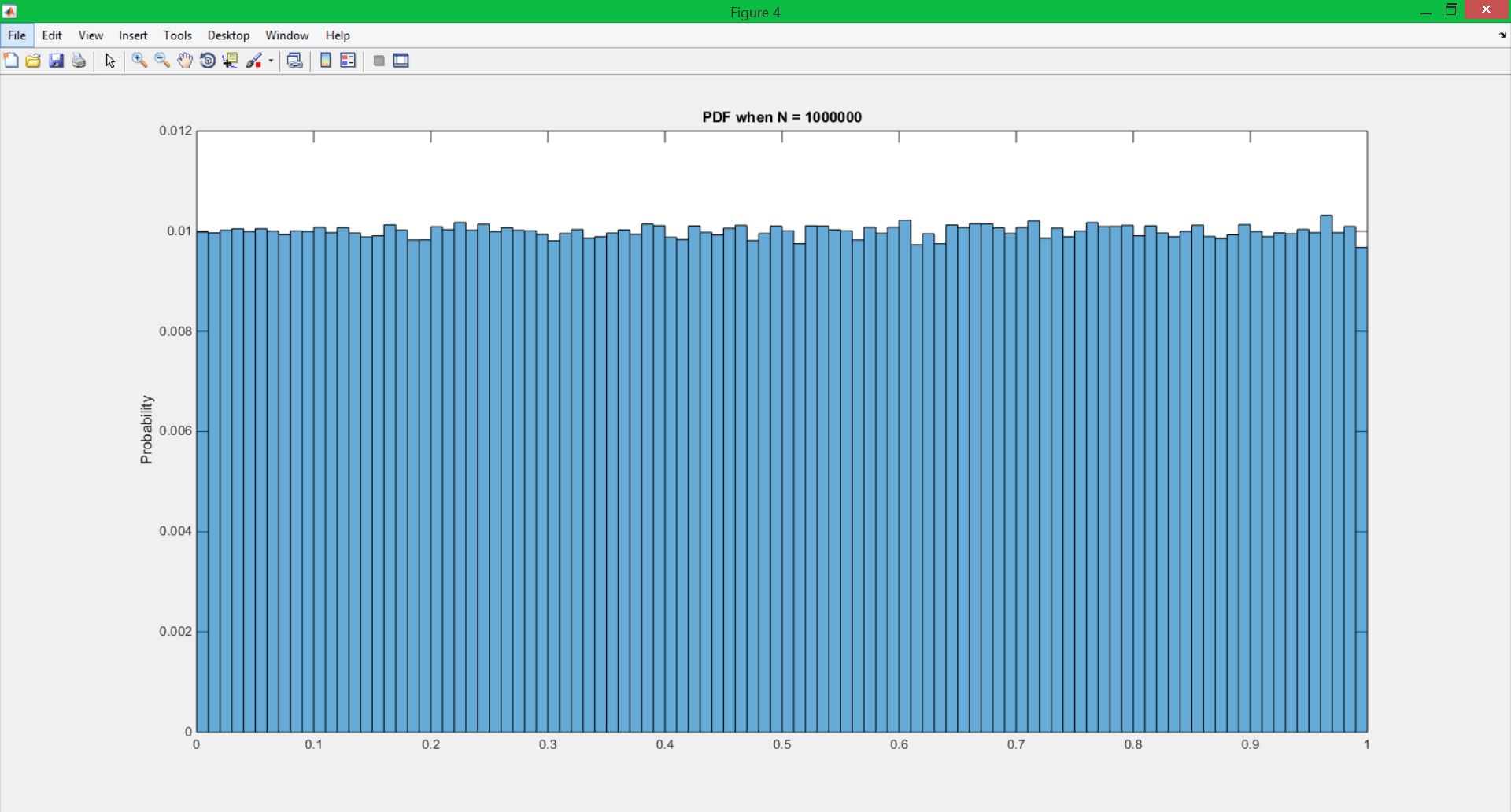
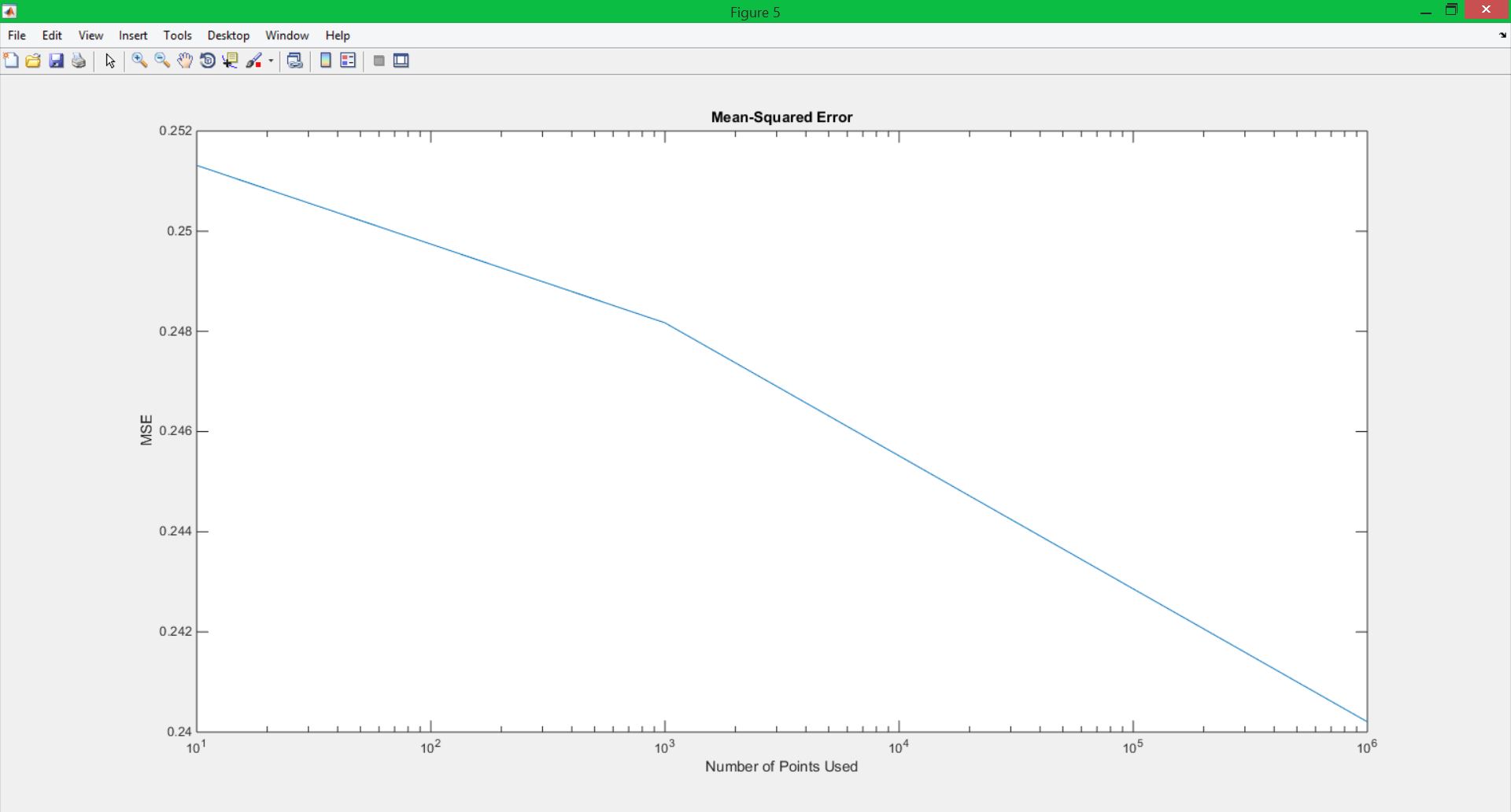


Figure 4



Finally, figure 5 shows the MSE of each signal.

Figure 5



# MATLAB Code

clear all;

close all; clc;

%Generate random numbers and set up vectors to be used

N = randi([1,10^6],1,10^3);

x\_mu = zeros(1,length(N));

x\_sigmaX = zeros(1,length(N));

sigmaX = 1/12;

mu = 0.5;

%Find the mean and variance calculated using different points

%from N. Store in another vector

for i = 1:length(N)

[N\_mu, N\_sigmaX] = get\_Mean\_and\_Var(N(i));

x\_mu(i) = N\_mu;

x\_sigmaX(i) = N\_sigmaX;

end

%fid the errors

MuE = x\_mu - mu;

SigmaXE = x\_sigmaX - sigmaX;

%plot everythng using a scatter plot

figure(1);

subplot(2,1,1);

semilogx(N,MuE,'k.');

title('Error in Mean vs N Points');

xlabel('Number of points used in calculation');

ylabel('Mean Error');

subplot(2,1,2);

semilogx(N,SigmaXE,'r.');

title('Error in Variance vs N Points');

xlabel('Number of points used in calculations');

ylabel('Variance Error');

%Find MSE using vector N2

N2 = [10 1000 10^6];

MSEn = zeros(1,length(N2));

k = 2;

for i = 1:length(N2)

[MSEn(i),k] = meanSquaredE(N2(i),k);

end

%Plot MSE vs points used

figure(k);

semilogx(N2,MSEn);

title('Mean-Squared Error');

ylabel('MSE');

xlabel('Number of Points Used');

**The code first creates a vector of 1000 random sizes to be used to create the random signals. The sizes are passed into a function one at a time, which returns the variance and mean for the random signal of that size. Each mean and variance is stored in another vector. The errors is then calculated from these vectors and plotted. The same methodology is used for the second part, but the function this time estimates the pdf’s through histograms and calculates and returns the MSE, which is then stored in a vector. The *k* variable is used for defining figures inside the function.**

# Conclusions

This assignment was fairly intuitive. In part 1, as the number of samples in a signal increase, the error in both the man and variance will approach 0. This can easily be seen in the first figure. For the next problem, we see that with a low number of samples, not every bin is filled. This again is intuitive as if there are less samples than bins, there are not enough samples to fill each and every bin. As the number of samples increase, two things can be observed. One is that all the bins become filled, the other is that the pdf will begin to approach the expected value, which in the uniform case is that each value for the random variable has approximately the same probability. This convergence to the distribution makes sense because as each sample is taken, its probability is based on the pdf of the distribution. So as more samples are taken, the more the pdf will resemble the distribution. Finally, for the MSE, we see that the error falls as the number of samples increases. This agrees with the result found in the first problem, where increasing the number of samples minimizes the error. Since the MSE calculates the average error squared, it is relative to the error itself, and as the error decreases, so will the MSE>