Miles A. Vendetti-Houser

ECE 3522: Stochastics

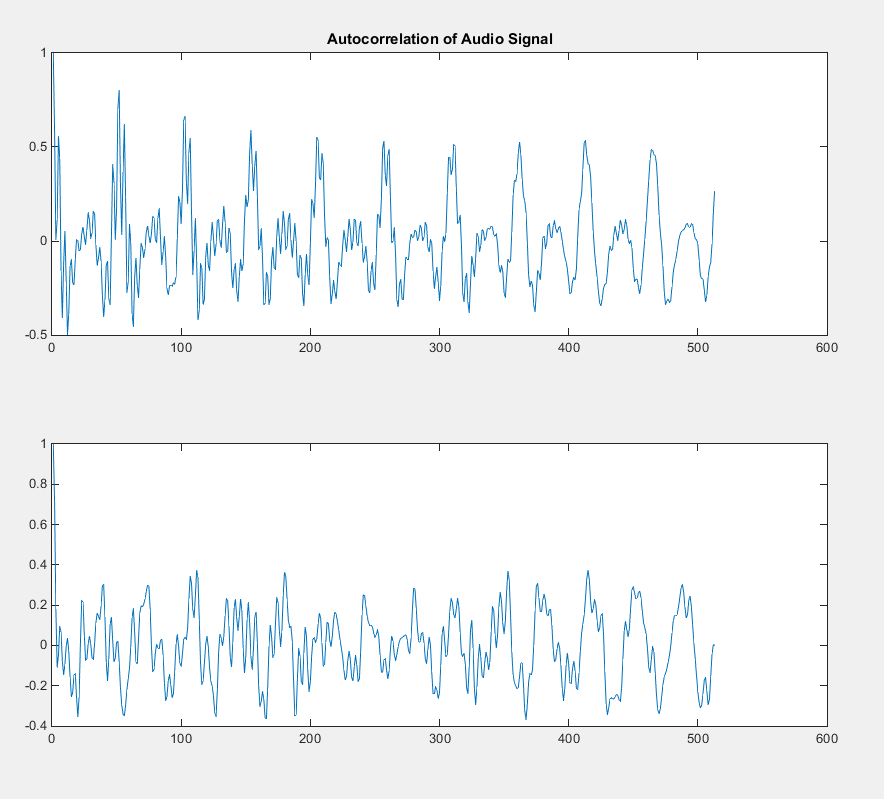
Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA 19122

# Problem Statement

The goal of this assignment is to introduce correlation and covariance, which is something mentioned throughout the lectures this semester.

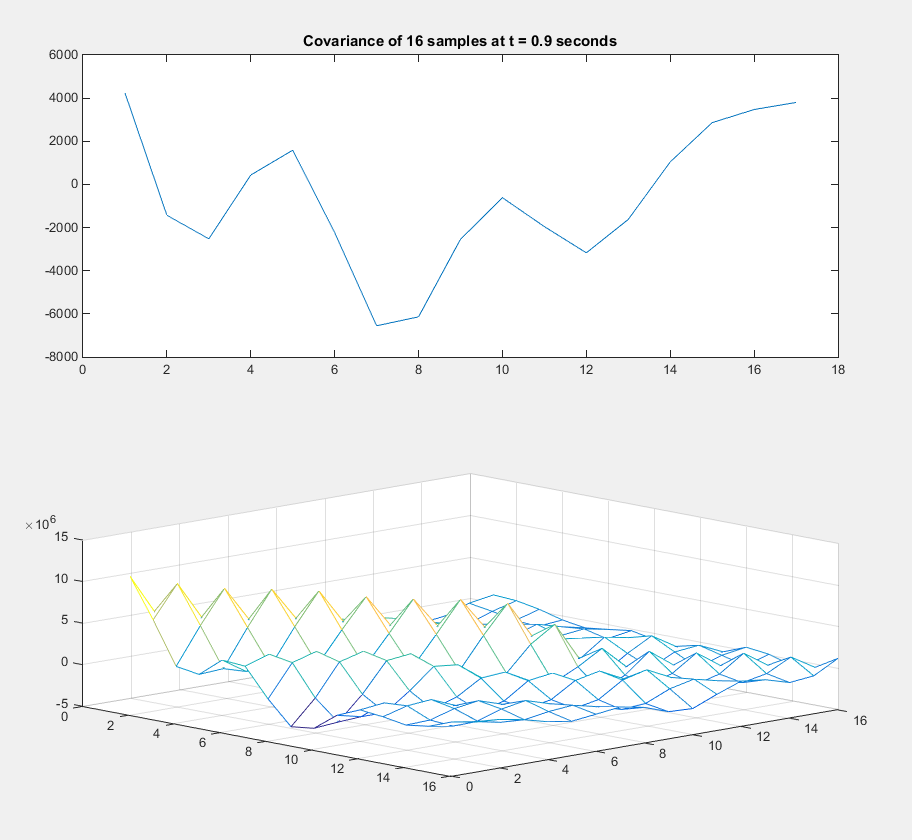
# Approach and Results

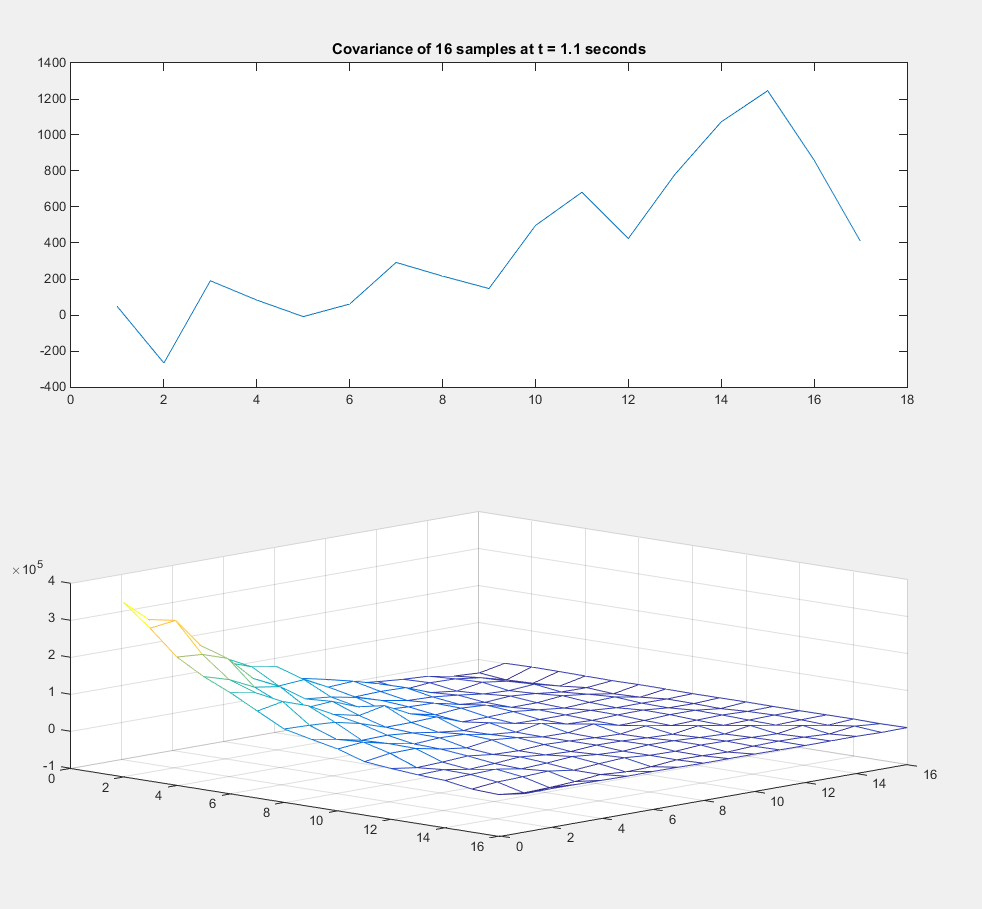
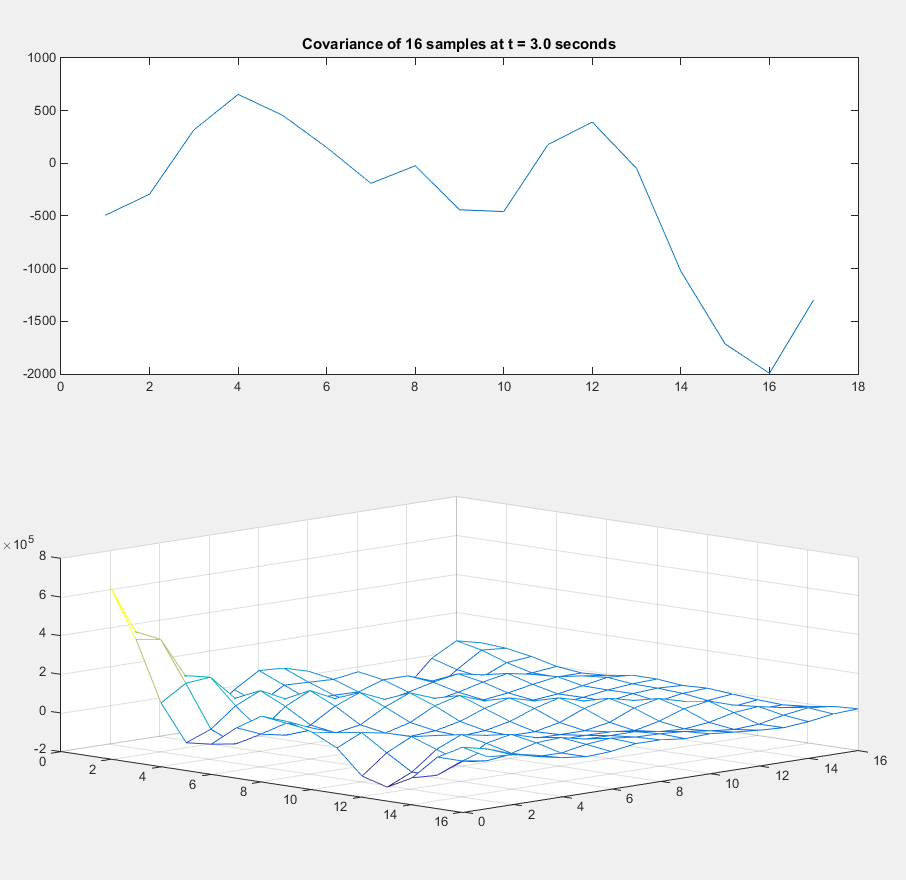
The assignment only requires the use of the audio signal. We start by simply importing the data into the workspace. The first thing to do was create a vector subset of data holding 240 data points from the original audio signal file starting from time, t=0.9 seconds. We find that the index for the t=0.9 must be 0.9\*8000 = 7200. Either way, I programmed my script well enough to take in the core parameters and output the desired x vector that takes the values of the original audio signal from 7200:7440. Next, a second vector, y, was created to contain 240 samples shifted by k samples. Using a *for* loop to iterate k starting from 1 up to 512, the correlation was computed between both vectors, x and y, and stored in a new vector called R1. This was repeated using a starting time of t=3.0. The resulting plot is shown below.



The subplot on the top contains correlation values very close to one. Without looking at the original graph of the audio signal, one can assume that k hits values shifting y to result in periodic correlation values. The audio signal has a periodic tendency. Also, when *k* grows larger, the correlation becomes less rigid while diminishing in amplitude, but maintaining a periodic property. The bottom subplot was created using t = 3.0 and looping *k* from 1 to 512. The maximum correlation is around 0.4 telling us that from 3.0 seconds and on, the two vectors x and y contain low correlated values. This does have a slightly noisier composition to it compared to plot above and does have a periodic property as well.

Part two of the computer assignment asks us to start once again at t=0.9 seconds and create a vector with the first 16 samples. Computing the covariance using 240 samples, I decided to plot the results in order to provide a better graphical understanding of what was happening. This part was to be completed two more times using t=1.1 seconds and t=3.0 seconds. All figures are displayed below respectively.





# MATLAB Code

% Computer Assignment 05

% Stochastics [ECE 3522] Dr. Picone

clear;clc;clf;close all;

% import the audio signal

filename = 'rec\_01\_speech.raw';

ff1 = fopen(filename, 'r');

sig = fread(ff1, inf, 'int16');

fclose(ff1);

% Part one -- CORRELATION

%

% Start at t = 0.9 sec, freq\_s = 8000

%

% Declare function pointers for dynamic coding :)

% time shift...sample frequency...function handlers...

N = 240;

fs = 8000;

start = @(t) t\*fs;

LEFT = @(t) start(t);

RIGHT = @(t) LEFT(t) + N;

% define a vector of 240 datapoints

%

x = @(t) sig(LEFT(t):RIGHT(t),:);

% define vector of 240 samples

% containing samples shifted by k

y = @(t,k) sig((LEFT(t)+k) : (RIGHT(t)+k) ,:);

% Loop from k = [0, 512] and plot

%

t1 = 0.9;

for k=0:512

R1(k+1,:)=corr(x(t1), y(t1, k));

end

%plot

figure(1)

subplot(2,1,1)

plot(R1)

title('Autocorrelation of Audio Signal')

% Repeat for t = 3.0

%

t2 = 3.0;

for k = 0:512

R2(k+1,:)=corr(x(t2), y(t2,k));

end

% plot

subplot(2,1,2)

plot(R2)

%%

% Part two -- COVARIANCE

%

% Start at 0.9 seconds, take first 16 samples

%

% cov = ( ( 1/length(x) ) \* sum( x(n-i)\*x(n-j) ) )

clc;clf;close all;

t09 = 0.9;

t11 = 1.1;

t30 = 3.0;

N1= 15;

x09 = sig( (t09\*fs):(t09\*fs) + N1 ,: );

x11 = sig( (t11\*fs):(t11\*fs) + N1 ,: );

x30 = sig( (t30\*fs):(t30\*fs) + N1 ,: );

cov09 = zeros([N1+1 N1+1]);

cov11 = zeros([N1+1 N1+1]);

cov30 = zeros([N1+1 N1+1]);

sum09 = 0;

for i=0:N1

for j=0:N1

sum09 = 0;

sum11 = 0;

sum30 = 0;

for n = 1:N1+1

if (n-j <= 0 || n-i <= 0)

result09 = 0;

result11 = 0;

result30 = 0;

else

result09 = x09(n-i) .\* x09(n-j);

sum09 = sum09 + result09;

result11 = x11(n-i) .\* x11(n-j);

sum11 = sum11 + result11;

result30 = x30(n-i) .\* x30(n-j);

sum30 = sum30 + result30;

end

end

cov09(i+1,j+1) = 1/N1 \*sum09;

cov11(i+1,j+1) = 1/N1 \*sum11;

cov30(i+1,j+1) = 1/N1 \*sum30;

end

end

figure(2)

subplot(2,1,1)

plot(sig(t09\*fs:t09\*fs+16))

title('Covariance of 16 samples at t = 0.9 seconds')

subplot(2,1,2)

mesh(cov09)

figure(3)

subplot(2,1,1)

plot(sig(t11\*fs:t11\*fs+16))

title('Covariance of 16 samples at t = 1.1 seconds')

subplot(2,1,2)

mesh(cov11)

figure(4)

subplot(2,1,1)

plot(sig(t30\*fs:t30\*fs+16))

title('Covariance of 16 samples at t = 3.0 seconds')

subplot(2,1,2)

mesh(cov30)

# Conclusions

The two covariance matrices differ in their data because the data is calculated over a different range of data. The covariance matrix for t = 1.1 secs has a smaller periodic tendency over the span of the 16 points, while the covariance for t=3.0 seconds computes the points which experience more replicated information. Both of the plots converge to zero at the tail-end of the matrix because the two points are nothing alike!