ABDURRAHİM ESKİN- 2016400387

CMPE 362: SIGNAL PROCESSING

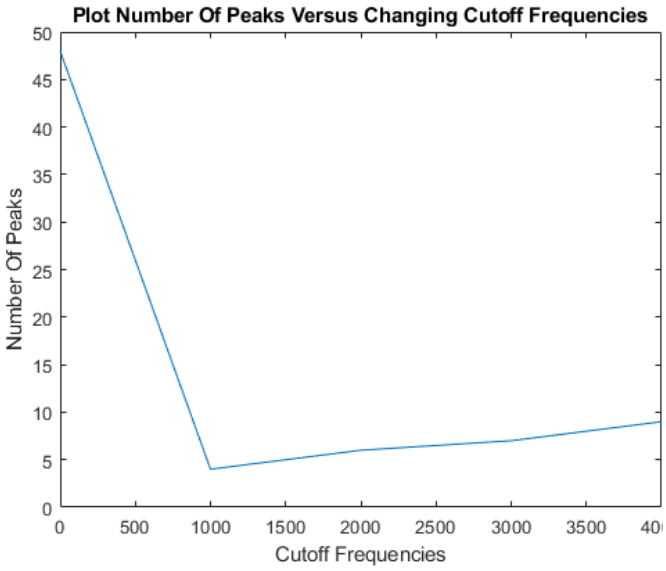
Spring 2018

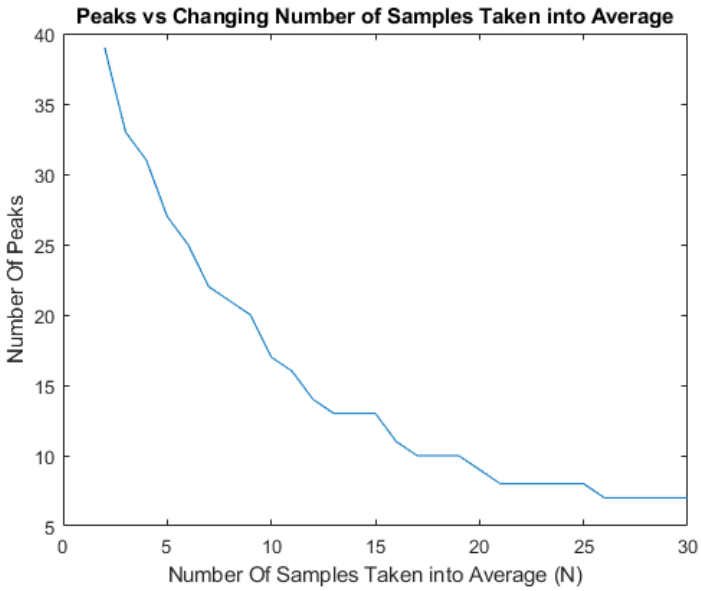
1-)In the first part, I implemented IIR lowpass filter with different cutoff frequencies to the exampleSignal from the first homework, in the first part I kept the information of the peaks versus changing filter frequencies after that i plot both peaks numbers and frequencies in the figure and showed changes in their relations.

%read the wave file

```
wfile=('exampleSignal.csv');
M=csvread(wfile, 1, 0);


[pks0,locs]=findpeaks(M, 10, 'MinPeakProminence', 1.10)
a0=numel(pks0);

%LOWPASS FILTER IMPLEMENTATION
%conform the lowpass with different limit frequencies and kept number of
%peaks according to changing filter frequencies


lpFilt1 = designfilt('lowpassiir','FilterOrder',8, ...
         'PassbandFrequency',1e3,'PassbandRipple',0.2, ...
         'SampleRate',200e3);

dataIn = M;
dataOut = filter(lpFilt1,dataIn);

[pks1,locs]=findpeaks(dataOut, 10, 'MinPeakProminence', 1.10);

a1 = numel(pks1);


lpFilt2 = designfilt('lowpassiir','FilterOrder',8, ...
         'PassbandFrequency',2e3,'PassbandRipple',0.2, ...
         'SampleRate',200e3);

dataIn = M;
dataOut = filter(lpFilt2,dataIn);
[pks2,locs]=findpeaks(dataOut, 10, 'MinPeakProminence', 1.10);

b = numel(pks2);




lpFilt3 = designfilt('lowpassiir','FilterOrder',8, ...
         'PassbandFrequency',3e3,'PassbandRipple',0.2, ...
         'SampleRate',200e3);

dataIn = M;
dataOut = filter(lpFilt3,dataIn);
```

```
[pks3,locs]=findpeaks(dataOut, 10, 'MinPeakProminence', 1.10);

c = numel(pks3);



lpFilt4 = designfilt('lowpassiir','FilterOrder',8, ...
         'PassbandFrequency',4e3,'PassbandRipple',0.2, ...
         'SampleRate',200e3);

dataIn = M;
dataOut = filter(lpFilt4,dataIn);
[pks4,locs]=findpeaks(dataOut, 10, 'MinPeakProminence', 1.10);

d = numel(pks4);

x = [a0 a1  b  c  d] ;
y = [0 1000 2000 3000 4000];

figure;
plot(y,x);
title('Plot Number Of Peaks Versus Changing Cutoff Frequencies');
ylabel('Number Of Peaks');
xlabel('Cutoff Frequencies');


%MOVING AVERAGE FILTER IMPLEMANTATION
%I have created moving average filter to show chancing N number I used for
%loop and I plotted number of peaks versus changing N in the figure

a = 1;
dataIn = M;
pkse = 1:29 ;
counter1 = 1:29;
for n  = 2:30
    h = ones(1,n)/ n;
    yma = filter(h, a, dataIn);
    [pks6,locs]=findpeaks(yma, 10, 'MinPeakProminence', 1.10);
    pkse(n-1) = numel(pks6);
end

for n = 2:30
    counter(n-1)= n;
end

figure;
plot (counter,pkse);
title('Peaks vs Changing Number of Samples Taken into Average')
ylabel('Number Of Peaks')
xlabel('Number Of Samples Taken into Average (N)')
```





2-) None of the sounds are the same. Because in the first two exercises we have different set of samples than the last two. Also the last two have different frequencies, so none of them are the same.

```
%% EXERCISE I
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Re-arrange the data so that   %
%   the frequency is quadrupled and play the file   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
sound(y(1:4:end), Fs);

duration = numel(y) / Fs;
pause(duration)



%% EXERCISE II
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Re-arrange the data so that   %
%   the frequency is halved and play the file  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

sound(y(1:0.5:end), Fs);

duration = numel(y) / Fs;
pause(duration * 2 + 2)



%% EXERCISE III
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Double Fs and play the sound  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

sound(y, Fs*2);
```

```
duration = numel(y) / Fs;
pause(duration)

%% EXERCISE IV
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Divide Fs by two and play the sound  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


sound(y, Fs/2);

duration = numel(y) / Fs;
```

3-) I wrote a 24 by 24 matrix which includes the 24 equations on the 9 points; 16 equations satisfying the 9 points, 7 equations satisfying the equality of derivations on joint internal points and one assumption that a_1 is zero thus the first equation is linear. I wrote another matrix(24 by 1) which holds the right hand sides of the 24 equations. I used linsolve to find the coefficients to the equations. I used these coefficients to create 8 equations and plotted them on corresponding intervals.

```
y = [1025, 1400, 1710, 2080, 2425, 2760, 3005, 2850, 2675];
x = [265, 400, 500, 700, 950, 1360, 2080, 2450, 2940];

A = [265^2 265 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
     400^2 400 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 400^2 400 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 500^2 500 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 500^2 500 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 700^2 700 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 700^2 700 1 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 950^2 950 1 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 950^2 950 1 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 1360^2 1360 1 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1360^2 1360 1 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2080^2 2080 1 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2080^2 2080 1 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2450^2 2450 1 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2450^2 2450 1;
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2940^2 2940 1];

b = [1025 1400 1400 1710 1710 2080 2080 2425 2425 2760 2760 3005 3005 2850
2850 2675];
Res = linsolve(A, b');

xx = 265:400;
p1 = xx.^2*Res(1)+xx*Res(2);
plot(xx,p1);
hold on;
xx = 400:500;
p2 = xx.^2*Res(4)+xx*Res(5);
```

```
plot(xx,p2);
hold on;
xx = 500:700;
p3 = xx.^2*Res(7)+xx*Res(8);
plot(xx,p3);
hold on;
xx = 700:950;
p4 = xx.^2*Res(10)+xx*Res(11);
plot(xx,p4);
hold on;
xx = 950:1360;
p5 = xx.^2*Res(13)+xx*Res(14);
plot(xx,p5);
hold on;
xx = 1360:2080;
p6 = xx.^2*Res(16)+xx*Res(17);
plot(xx,p6);
hold on;
xx = 2080:2450;
p7 = xx.^2*Res(19)+xx*Res(20);
plot(xx,p7);
hold on;
xx = 2450:2940;
p8 = xx.^2*Res(22)+xx*Res(23);
plot(xx,p8);
hold on;
```