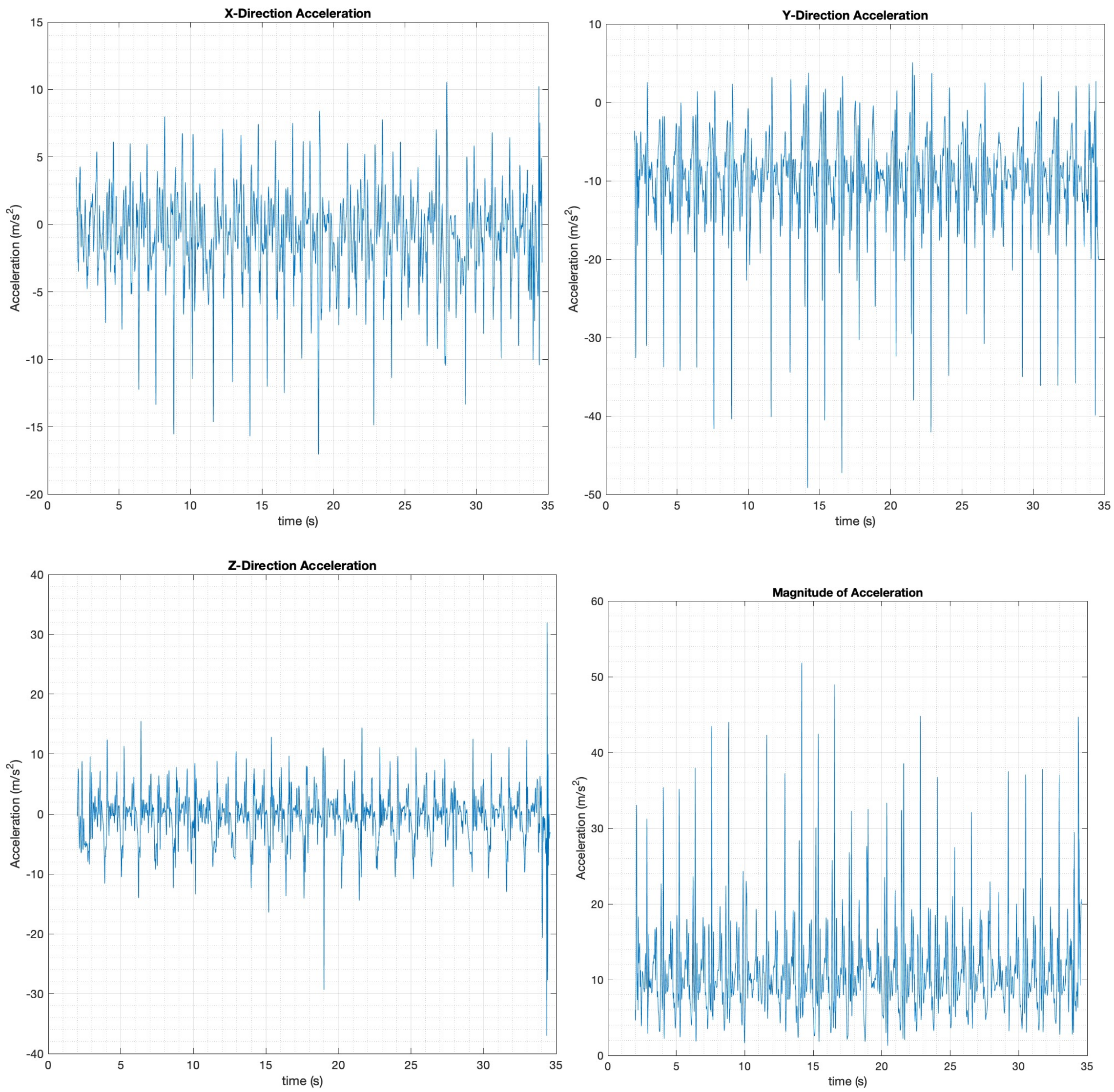
Zach Hoffman and Chris Osgood

ME220 Lab 1 Writeup
5/4/22

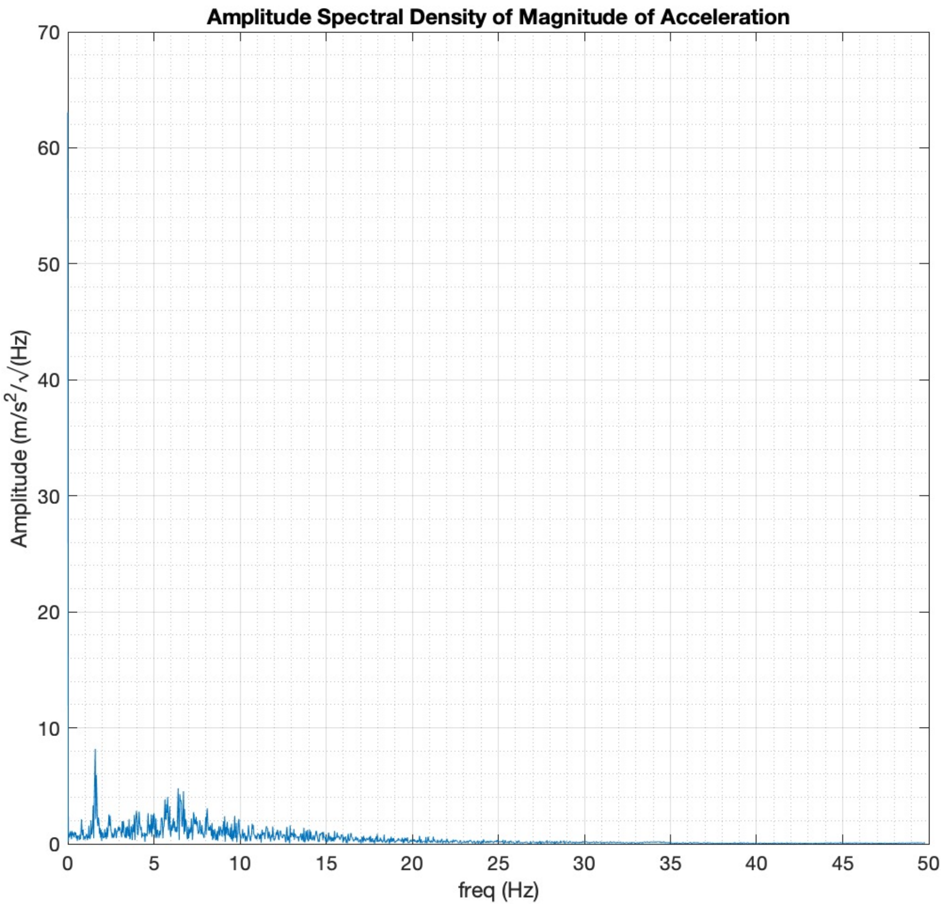
Part 1: A basic algorithm

1. Acceleration plot for x, y, z axis

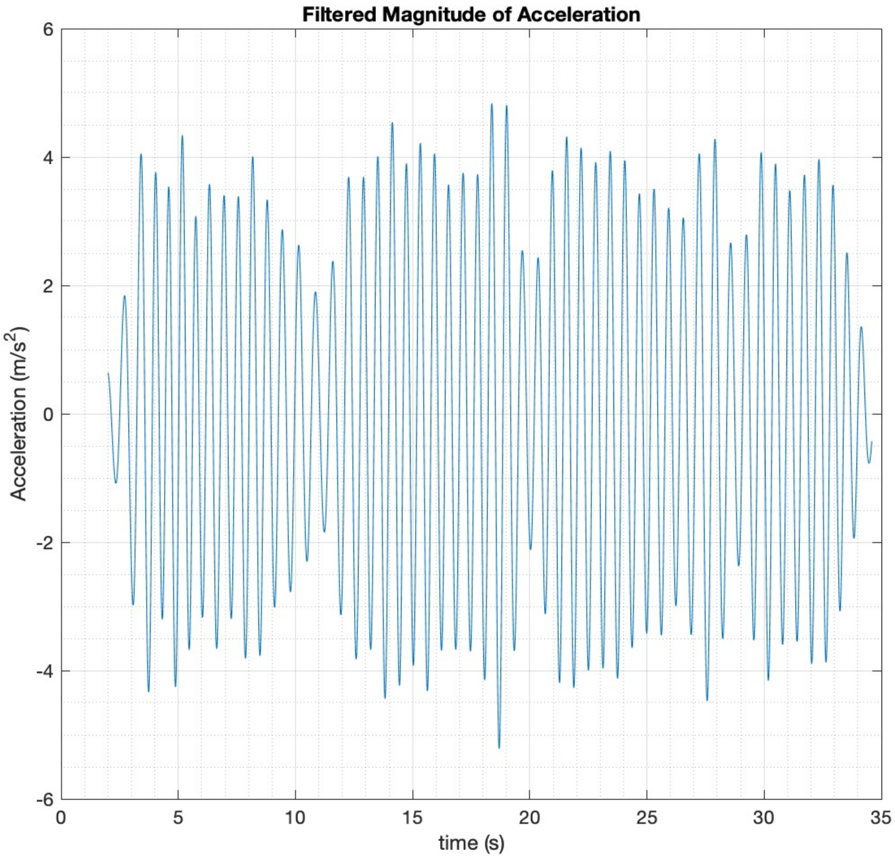


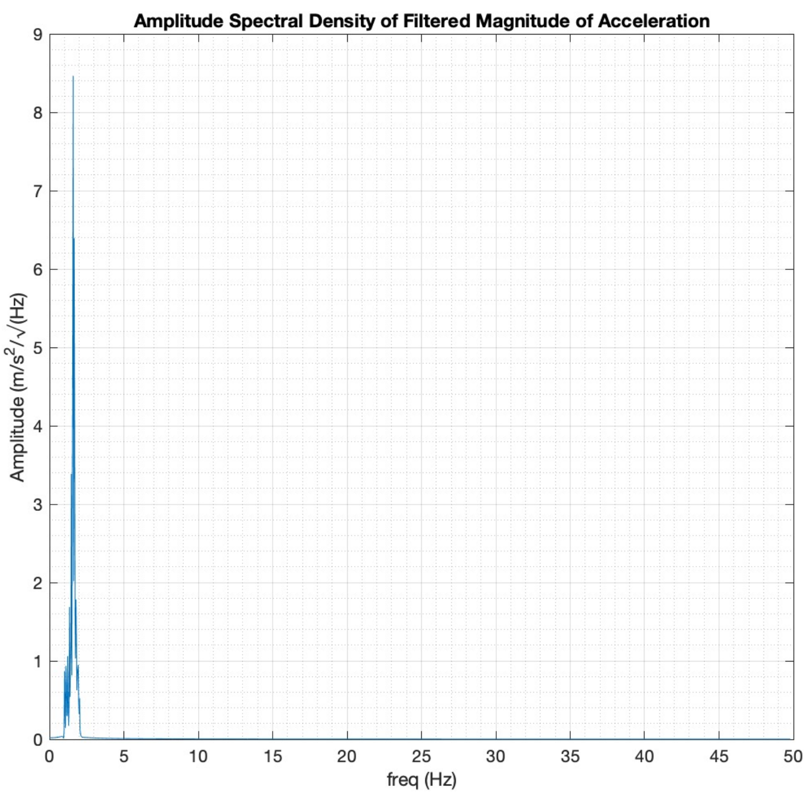
2. Acceleration plot for magnitude: There are spikes of >20 m/s^2, which are indicative of a step being taken. When we change direction, the magnitude of the step decreases by around 10 m/s^2. These spikes seem to occur about once or twice every second. Each positive spike has a corresponding negative spike right after.
3. ASD plot: The step signals amplitude is around 8.2 m/s^2 around 1.6 Hz. Also notice the largest amplitude is the DC offset at zero frequency due to gravity. And there are some

higher frequency amplitudes as well, which are likely harmonics, noise, other movements, etc.



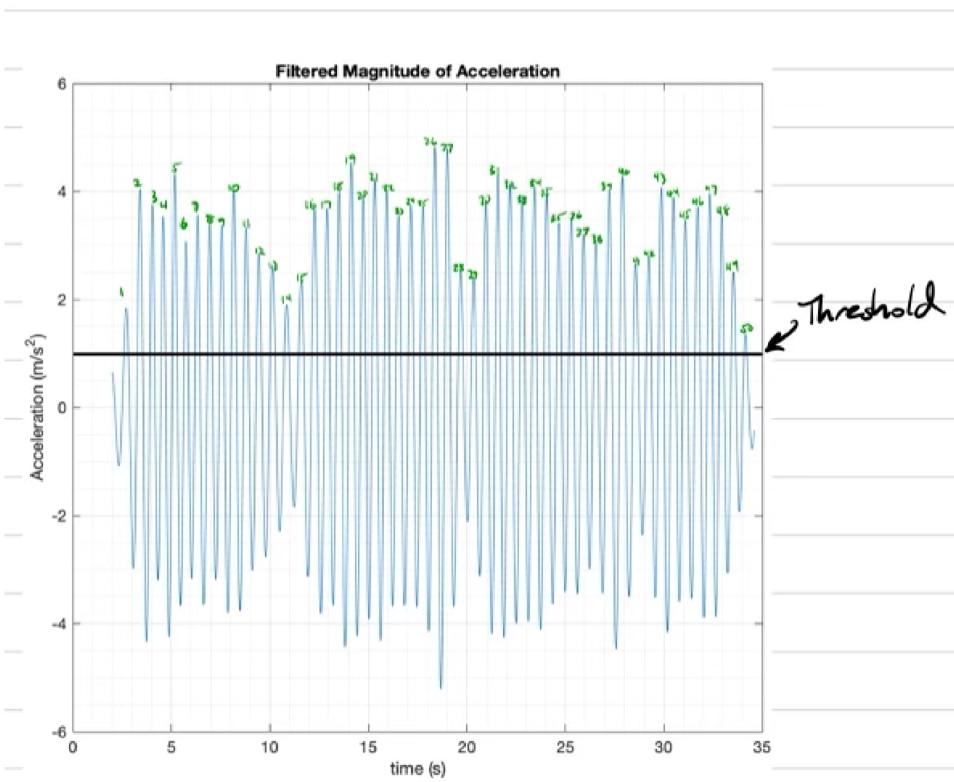
4. We used a bandpass filter whose cutoff frequencies are at 1 and 2 Hz. We chose a bandpass filter to eliminate high frequency noise and DC biasing (such as acceleration due to gravity), and we chose the corner frequencies from our visual estimate of walking frequency, giving us room for different walking speeds. Time and frequency domain plot of filtered Acceleration magnitude:


Filtered Magnitude of Acceleration
Acceleration (m/s^2)
time (s)
6
4
2
0
-2
-4
-6
0
5
10
15
20
25
30
35


Amplitude Spectral Density of Filtered Magnitude of Acceleration
Amplitude (m/s^2/√(Hz)
freq (Hz)
9
8
7
6
5
4
3
2
1
0
0
5
10
15
20
25
30
35
40
45
50

5. We chose a threshold of 1.5 m/s^2 off of our filtered magnitude of acceleration plot. Our algorithm simply looks for the threshold on a rising edge and adds it to a step counter.

6. Our algorithm identified 50 steps, which is the same number of steps that we counted.



7. Repeating the steps above 5 more times, we get very accurate results across the board. The algorithm counts steps pretty accurately for both of us, in different pockets, oriented differently while in a pocket, etc. If anything, the algorithm is either correct, or off by only one step. Of course, this is with nicely trimmed data.

Table of Contents

ME220, Lab 1

Chris Osgood and Zach Hoffman, 4/26/22

```
clc; clear; close all;

%%%%%%%% Part 1 %%%%%%%%

% pre-trimmed data
data = readmatrix('Accelerometer1.csv');
% data = readmatrix('Accelerometer2.csv');
% data = readmatrix('Accelerometer3.csv');
% data = readmatrix('Accelerometer4.csv');
% data = readmatrix('Accelerometer5.csv');

t = data(:,1);
accel = data(:,2:4);
```

plot 3 axis accelerations

```
figure;
plot(t, accel(:,1));
title('X-Direction Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
figure;
plot(t, accel(:,2));
title('Y-Direction Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
figure;
plot(t, accel(:,3));
title('Z-Direction Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

find and plot magnitude of accelerations

```
mag = vecnorm(accel, 2, 2);
```

```
figure;
plot(t, mag);
title('Magnitude of Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

find and plot ASD of magnitude of acceleration

```
freq = 1 / mean(diff(t));
[f, mag_asd] = ASD(mag, freq);
figure();
plot(f,mag_asd);
title('Amplitude Spectral Density of Magnitude of Acceleration');
xlabel('freq (Hz)'); ylabel('Amplitude (m/s^2/\surd(Hz)');
grid on; grid minor;
```

Filter (band pass between 1 and 2 Hz)

```
filt = bandpass(mag, [1,2], freq);
figure;
plot(t, filt);
title('Filtered Magnitude of Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

plot ASD of fitlered data

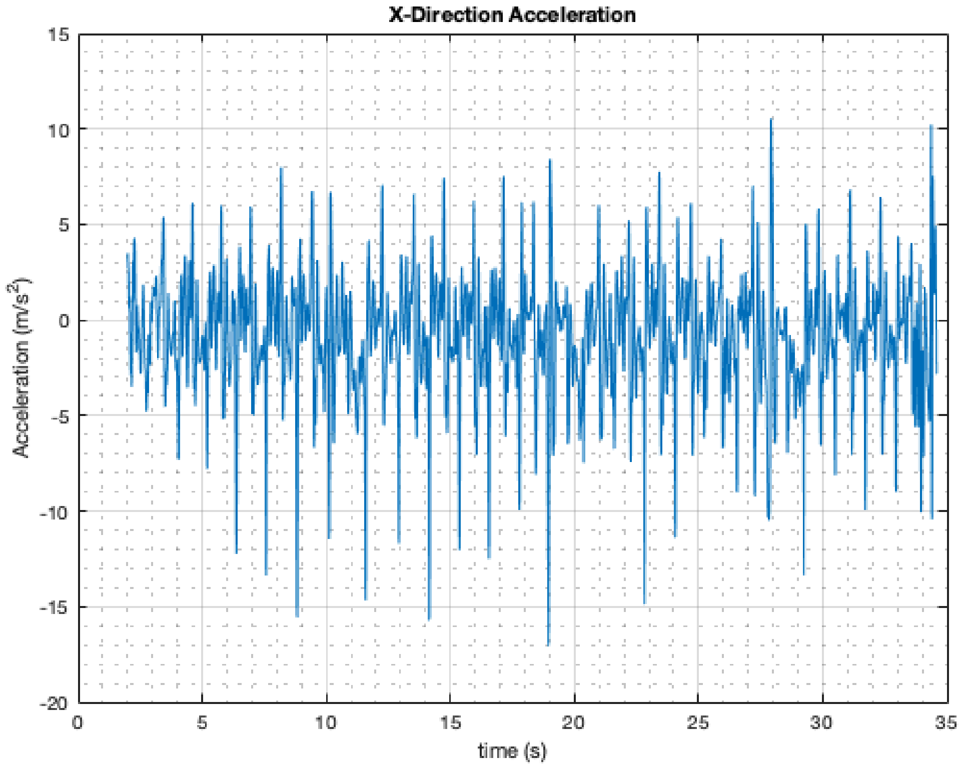
```
[f, filt_asd] = ASD(filt, freq);
figure;
plot(f, filt_asd);
title('Amplitude Spectral Density of Filtered Magnitude of
 Acceleration');
xlabel('freq (Hz)'); ylabel('Amplitude (m/s^2/\surd(Hz)');
grid on; grid minor;
```

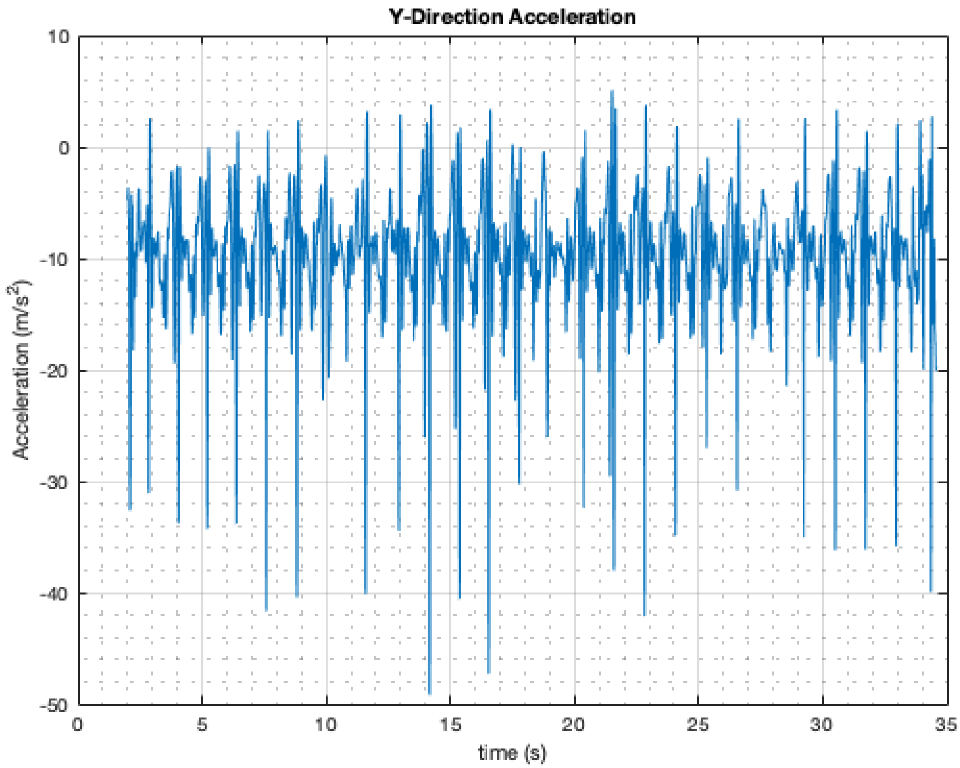
Count step threshold -- number of peaks over 1.5

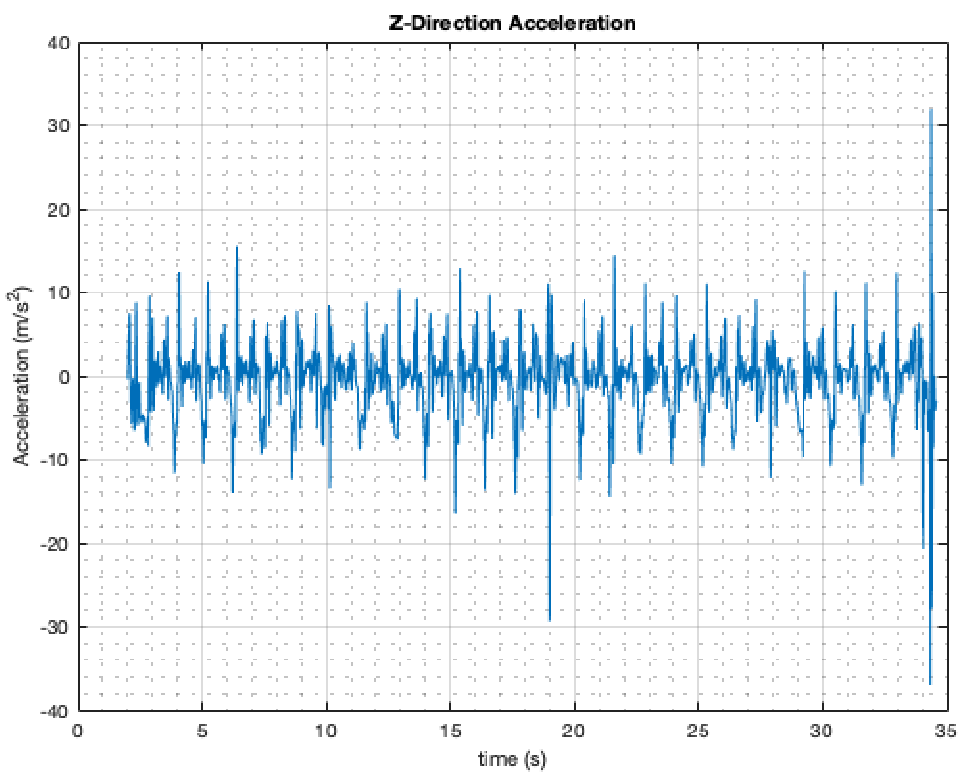
```
count = 0;
thresh = 1.5;
i = 1;
while i <= length(filt)
    if filt(i) > thresh
       count = count + 1;
       while filt(i) > thresh
           i = i + 10;
       end
    end
    i = i + 1;
```

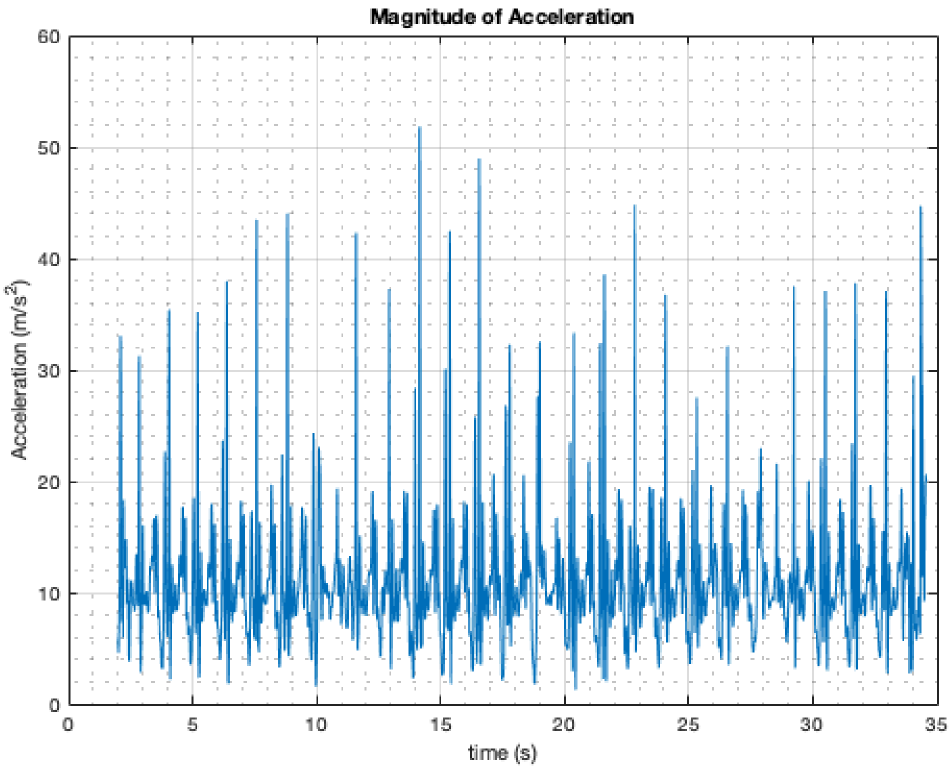
```
end
msg = "You must have taken " + count + " steps!";
disp(msg)


function [f, asd] = ASD(data, freq)
    % ASD uses fft to compute the amplitude spectral density of the
 data.
    % "data" should be a collumn vector in the time domain.
    % length(data) must be even.
    data_f = fft(data);
    L = length(data);
    data_f = data_f(1:L/2+1);
    psd = (1/(freq*L)) * abs(data_f).^2;
    psd(2:end-1) = 2*psd(2:end-1);
    asd = sqrt(psd);
    f = freq*(0:(L/2))/L;
end
```

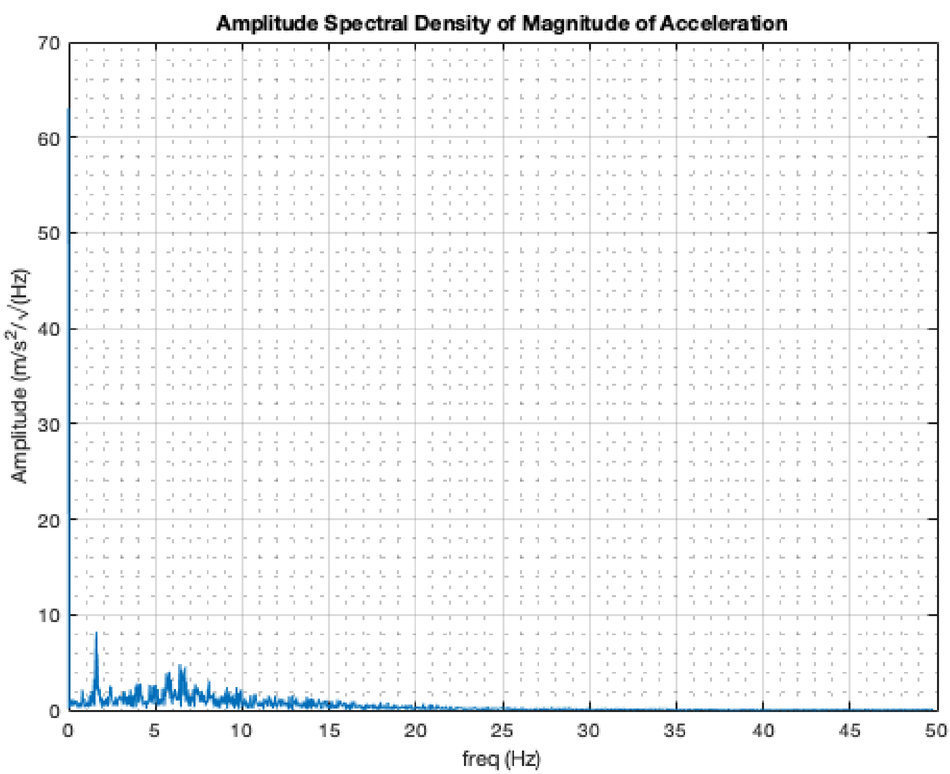
You must have taken 50 steps!

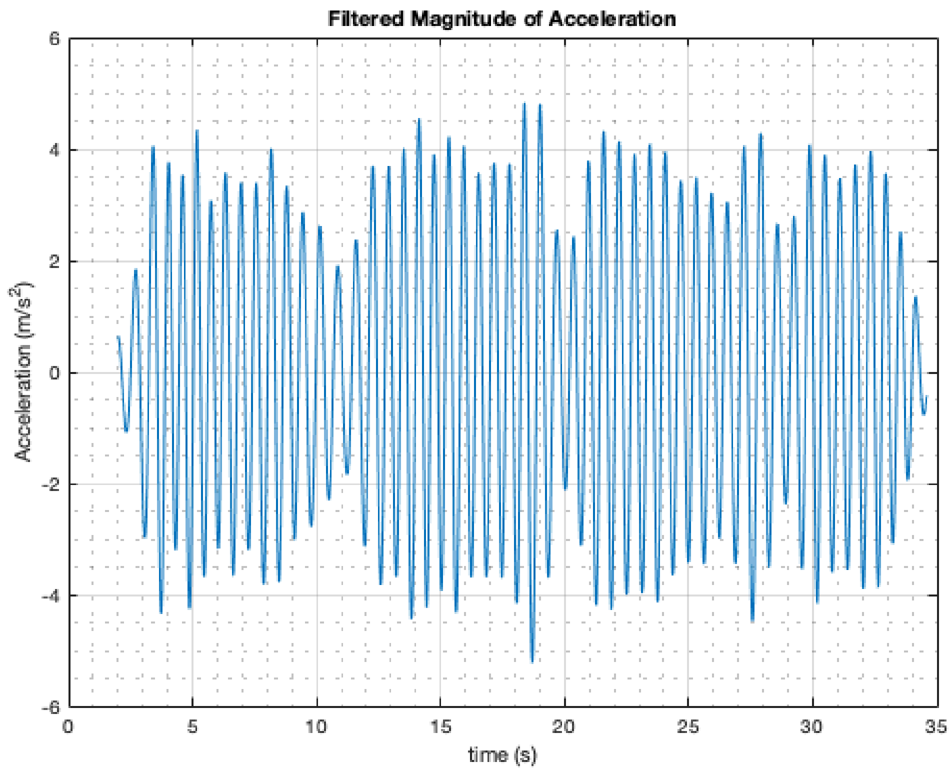


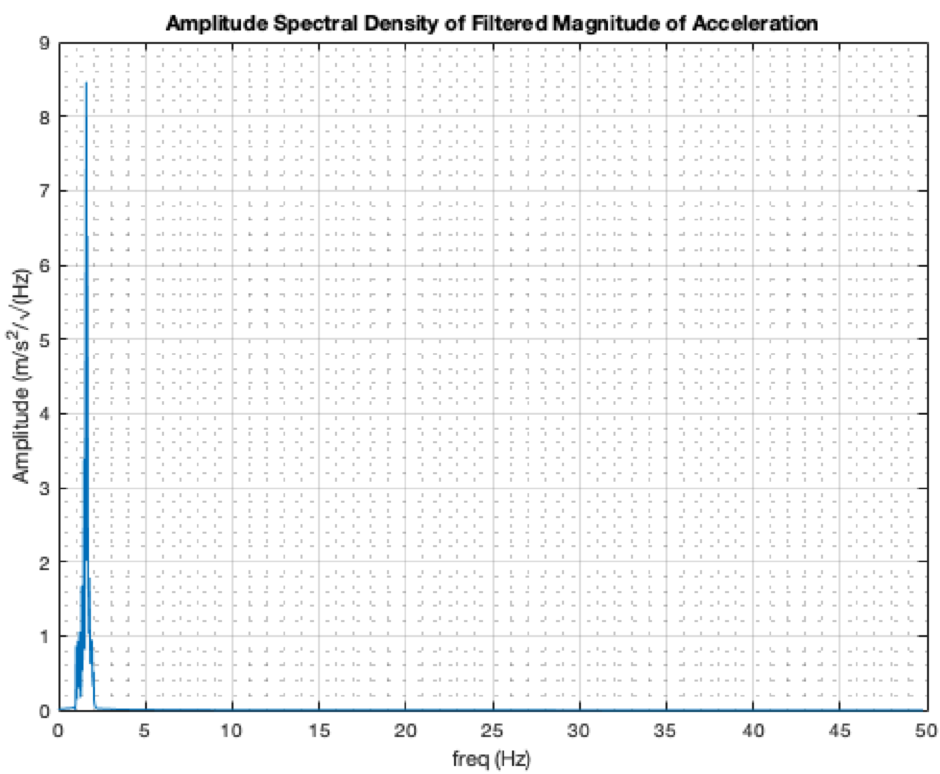

Y-Direction Acceleration
10
0
-10
-20
-30
-40
-50
Acceleration (m/s^2)
0
5
10
15
20
25
30
35
time (s)


Z-Direction Acceleration
40
30
20
10
0
-10
-20
-30
-40
Acceleration (m/s^2)
0
5
10
15
20
25
30
35
time (s)


Magnitude of Acceleration
Acceleration (m/s^2)
time (s)
0
5
10
15
20
25
30
35
40
50
60


Amplitude Spectral Density of Magnitude of Acceleration
Amplitude (m/s^2/√(Hz)
freq (Hz)
0
5
10
15
20
25
30
35
40
45
50
60
70

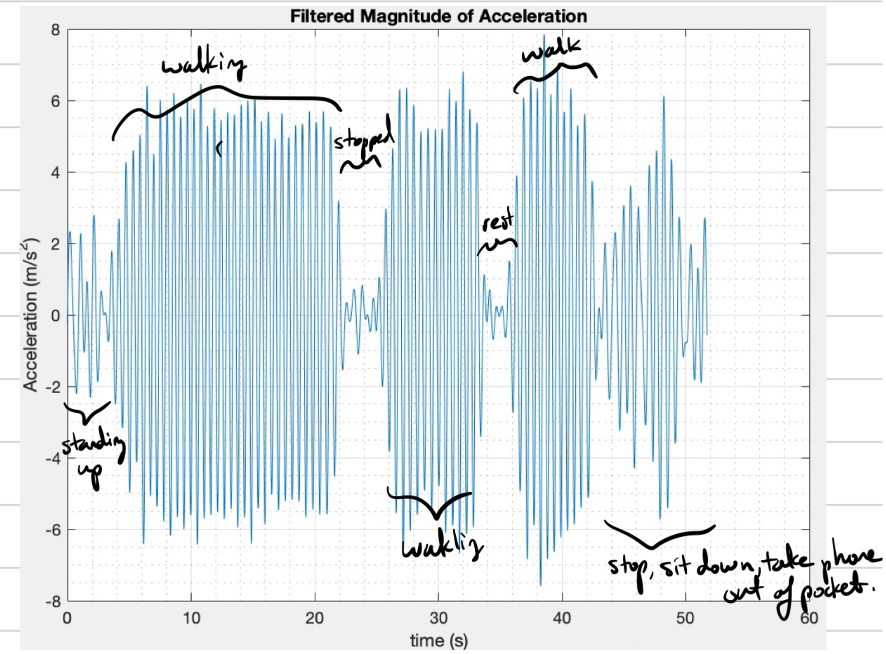

Filtered Magnitude of Acceleration
Acceleration (m/s^2)
time (s)
6
4
2
0
-2
-4
-6
0
5
10
15
20
25
30
35


Amplitude Spectral Density of Filtered Magnitude of Acceleration
Amplitude (m/s^2/√(Hz)
freq (Hz)
9
8
7
6
5
4
3
2
1
0
0
5
10
15
20
25
30
35
40
45
50

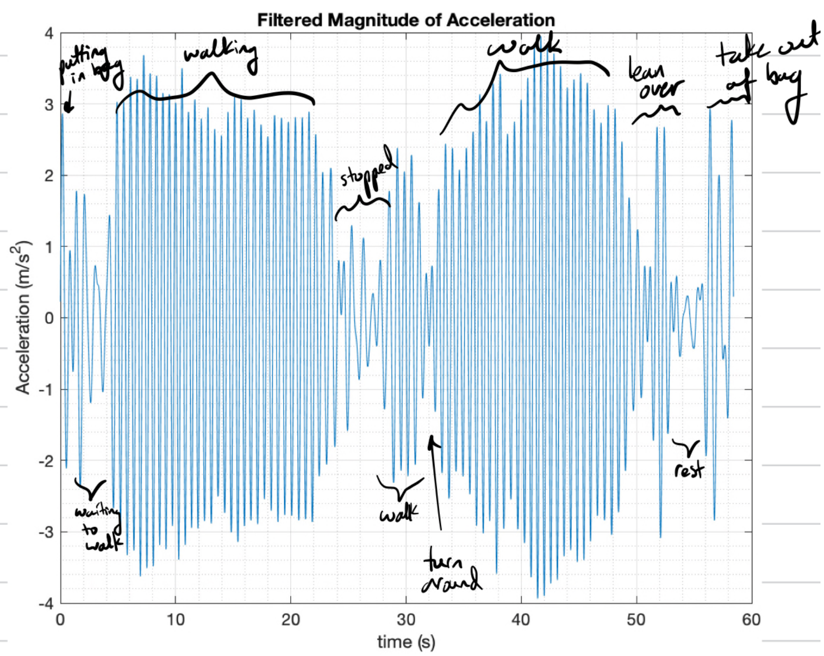
Published with MATLAB® R2020b

Part 2:

1. Did it.
2. Did it.
3. In pocket:



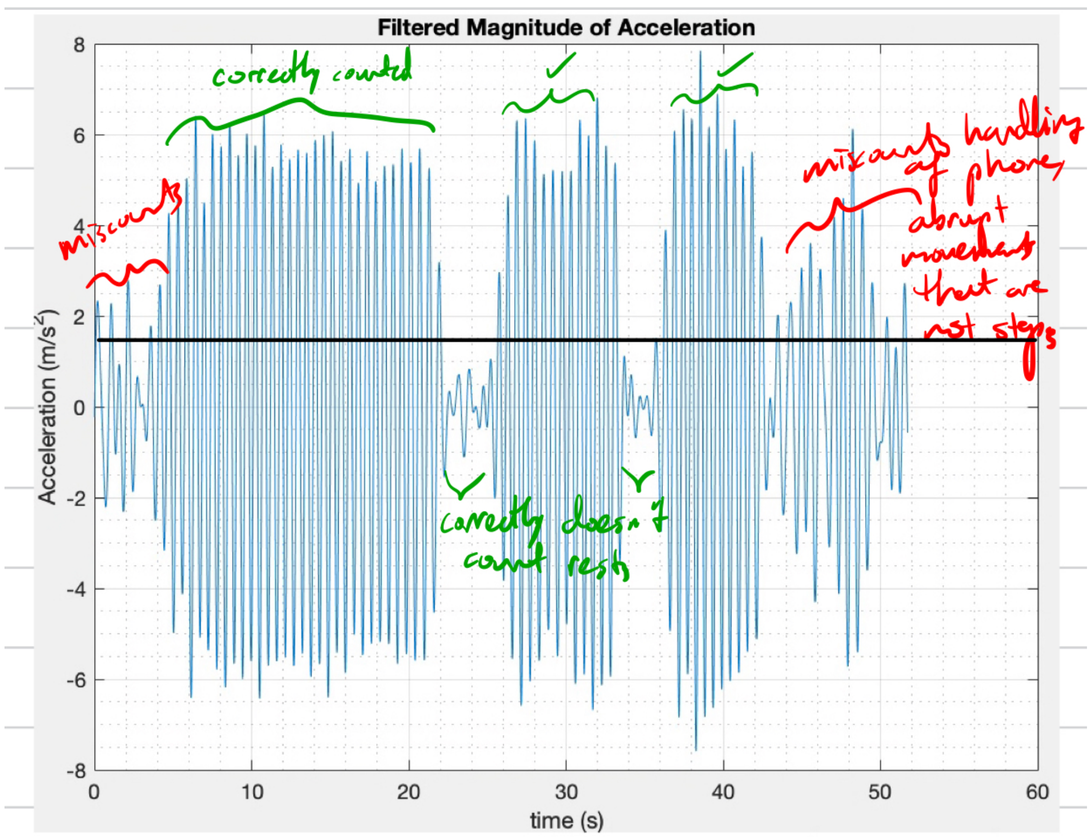
In bag:



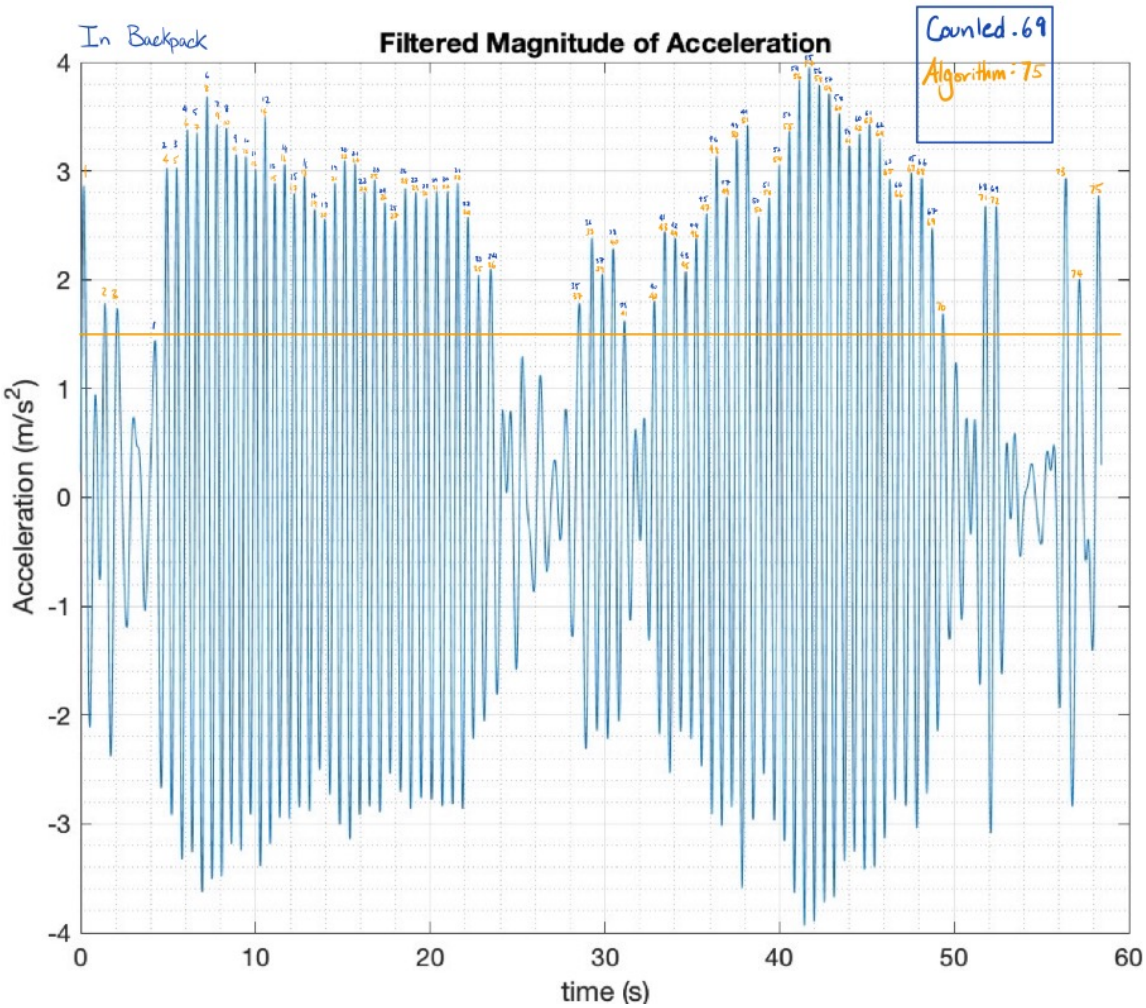
Resting/stopped is obvious to see since the filtered acceleration is much closer to zero, but other motions seem much more difficult to see. We may have to filter the data differently to distinguish them.

4. Analysis: Both in the backpack and in the pocket the algorithm overcounted. They counted handling of the phone and standing up and sitting down and leaning over all as steps. These present at least an **8.6% error** in the step count. We need to find a way to not count movements that appear as steps, but are not actually steps, i.e. subtract out non-steps. Plots below.

In Pocket (algorithm counts 76, should be 69)

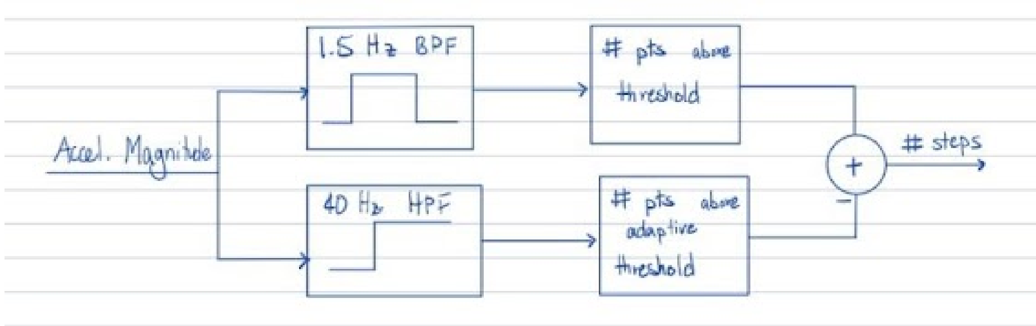


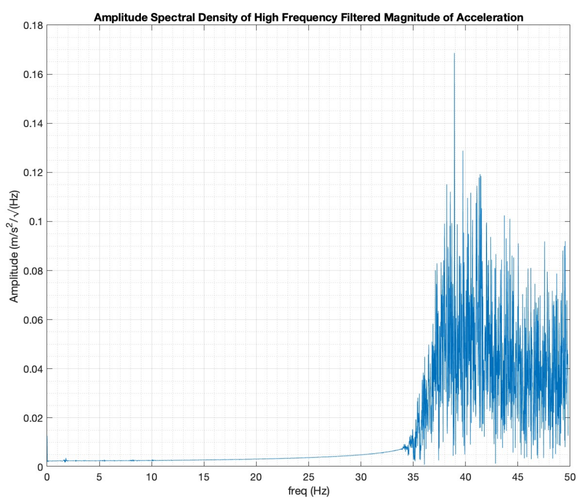
In Backpack:

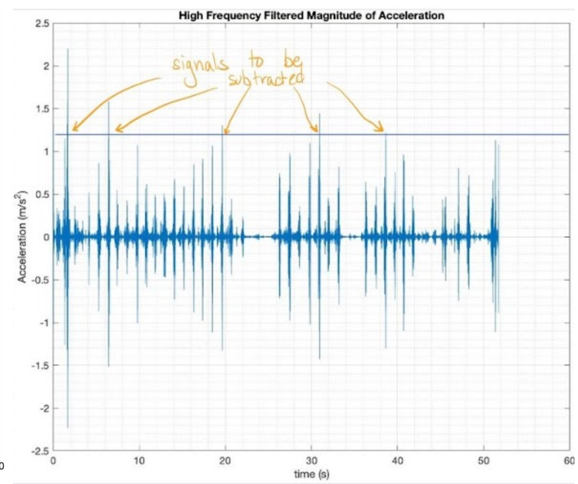


5. Our new algorithm implements two filters with adaptive threshold calculations. Looking at the data, we can see that many of the regions where our first algorithm counted jostling/non-stepping movement as steps also represented regions of high frequency signals. Our new implementation begins with the step counting algorithm described above. Then it applies a high pass filter on the raw data magnitude to isolate the regions of false step counting. It then determines the proper threshold, relative to the high-pass filtered data (0.5 * max_value). Finally, the algorithm subtracts out the number of data points that meet the threshold on a rising edge from the original step count

to get the final step count estimate.







6. Result: We see a significant improvement from the original algorithm and can do a much better job in filtering out non-step movements. The new algorithm counts 68 steps in the pocket. This is a decrease in error to **1.4%.** Both of these are significant improvements upon the original algorithm. So, yes the new algorithm performs better.
7. Result: In the bag, we also see a significant improvement from the original algorithm and can do a much better job in filtering out non-step movements. We now count 72 steps in the bag. This is a decrease in error to **4.3%.** We still overcount in the bag a bit, likely because high magnitude and high frequency jostles are damped by the bag, so our threshold on subtracting out steps isn't high enough to catch all the non-steps. Due to this reason, our algorithm

works slightly better in the pocket as opposed to in the bag. We then tested the algorithm a handful more times on un-trimmed data (both in a bag and in a pocket), and found that we got an error of less than 5% consistently.

Table of Contents

ME220, Lab 1

Chris Osgood and Zach Hoffman, 5/3/22


```
clc; clear; close all;

%%%%%%%% Part 2 %%%%%%%%

data = readmatrix('Accelerometer_23.csv'); % in pocket data
%data = readmatrix('Accelerometer_bag.csv'); % in bag data

t = data(:,1);
accel = data(:,2:4);
```

plot 3 axis accelerations

```
figure;
plot(t, accel(:,1));
title('X-Direction Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
figure;
plot(t, accel(:,2));
title('Y-Direction Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
figure;
plot(t, accel(:,3));
title('Z-Direction Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

find and plot magnitude of accelerations

```
mag = vecnorm(accel, 2, 2);
```

```
figure;
plot(t, mag);
title('Magnitude of Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

find and plot ASD of magnitude of acceleration

```
freq = 1 / mean(diff(t));
[f, mag_asd] = ASD(mag, freq);
figure();
plot(f,mag_asd);
title('Amplitude Spectral Density of Magnitude of Acceleration');
xlabel('freq (Hz)'); ylabel('Amplitude (m/s^2/\surd(Hz)');
grid on; grid minor;
```

try new centered bandpass

[max_asd, ind] = max(mag_asd(2:end)); f_c = f(ind); f_pass = [f_c-.3, f_c+.3];

Filter (band pass between 1 and 2 Hz)

```
f_pass = [1,2];
filt = bandpass(mag, f_pass, freq);
figure;
plot(t, filt);
title('Filtered Magnitude of Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

plot ASD of fitlered data

```
[f, filt_asd] = ASD(filt, freq);
figure;
plot(f, filt_asd);
title('Amplitude Spectral Density of Filtered Magnitude of
 Acceleration');
xlabel('freq (Hz)'); ylabel('Amplitude (m/s^2/\surd(Hz)');
grid on; grid minor;
```

Count step threshold -- number of peaks over 1.5

```
count = 0;
thresh = 1.5;
i = 1;
while i <= length(filt)
    if filt(i) > thresh
```

```
        count = count + 1;
        while filt(i) > thresh
            i = i + 10;
            if i > length(filt)
                break
            end
        end
    end
    i = i + 1;
end
msg = "Original Filter: You must have taken " + count + " steps!";
disp(msg)



%%%%%%% Updated Filter %%%%%%%%

%%%% Do a new filter to count high freq jostles. Then we will remove
 those
%%%% high freq jostles from our total step count to remove
 overcounting of
%%%% our steps when the original filter mistakes jostles for steps.
```

High pass filter, above 4 Hz

```
f_pass_hf = 40; % Hz
filt_hf = highpass(mag, f_pass_hf, freq);
figure;
plot(t, filt_hf);
title('High Frequency Filtered Magnitude of Acceleration');
xlabel('time (s)'); ylabel('Acceleration (m/s^2)');
grid on; grid minor;
```

plot ASD of hf fitlered data

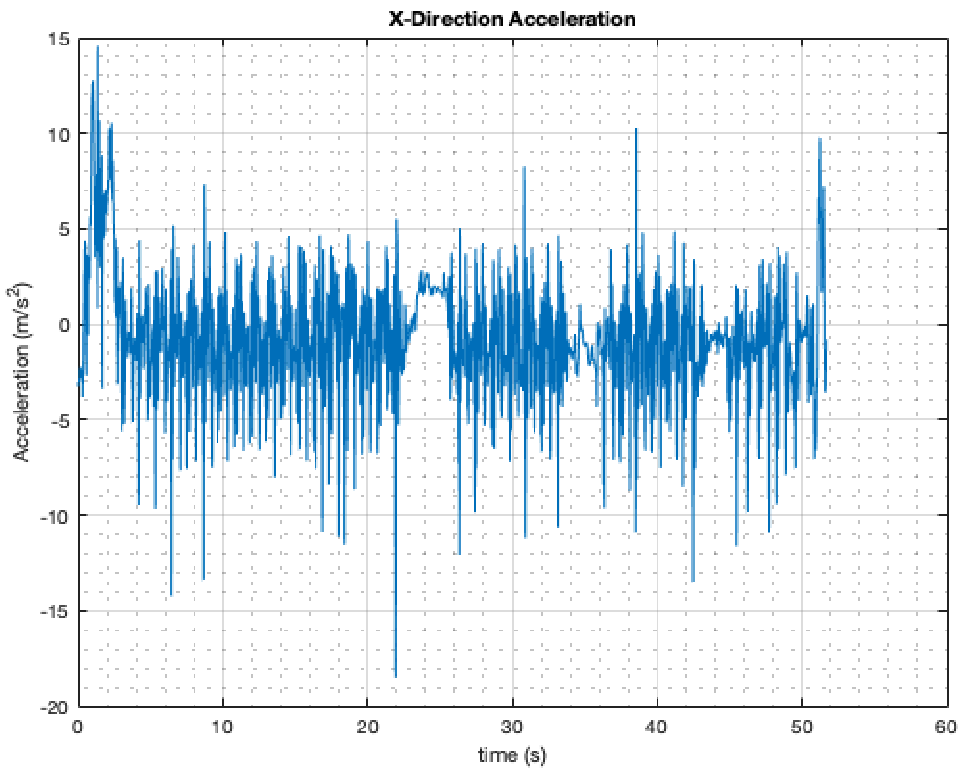
```
[f, filt_asd_hf] = ASD(filt_hf, freq);
figure;
plot(f, filt_asd_hf);
title('Amplitude Spectral Density of High Frequency Filtered Magnitude
 of Acceleration');
xlabel('freq (Hz)'); ylabel('Amplitude (m/s^2/\surd(Hz)');
grid on; grid minor;
```

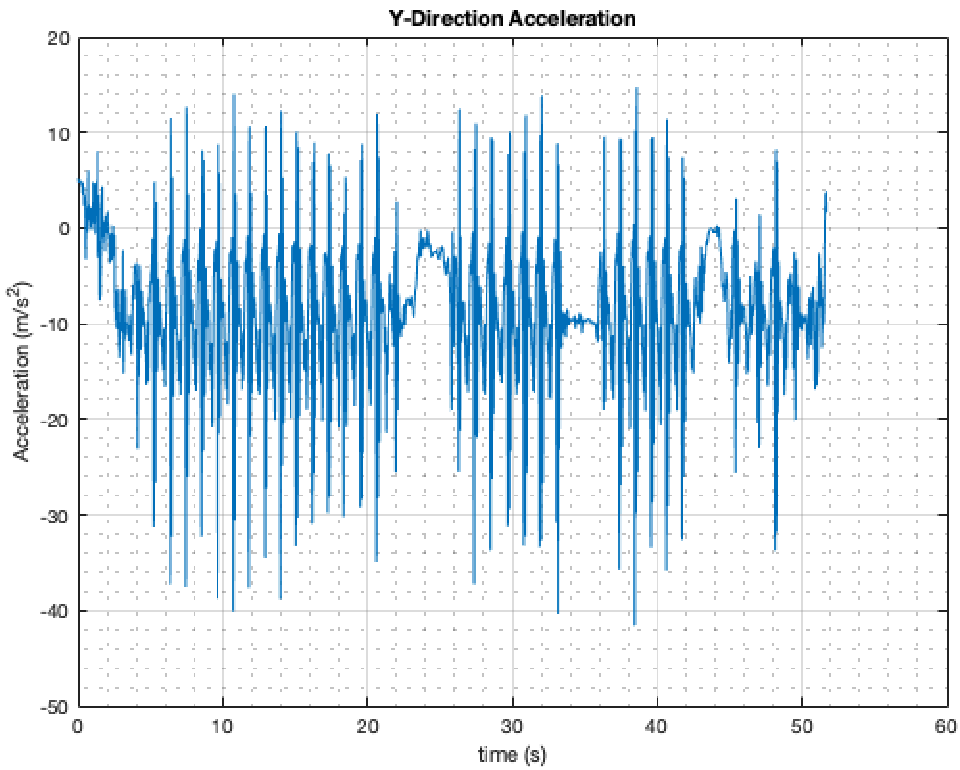
Count high freq jostles threshold as number of peaks over thresh_hf

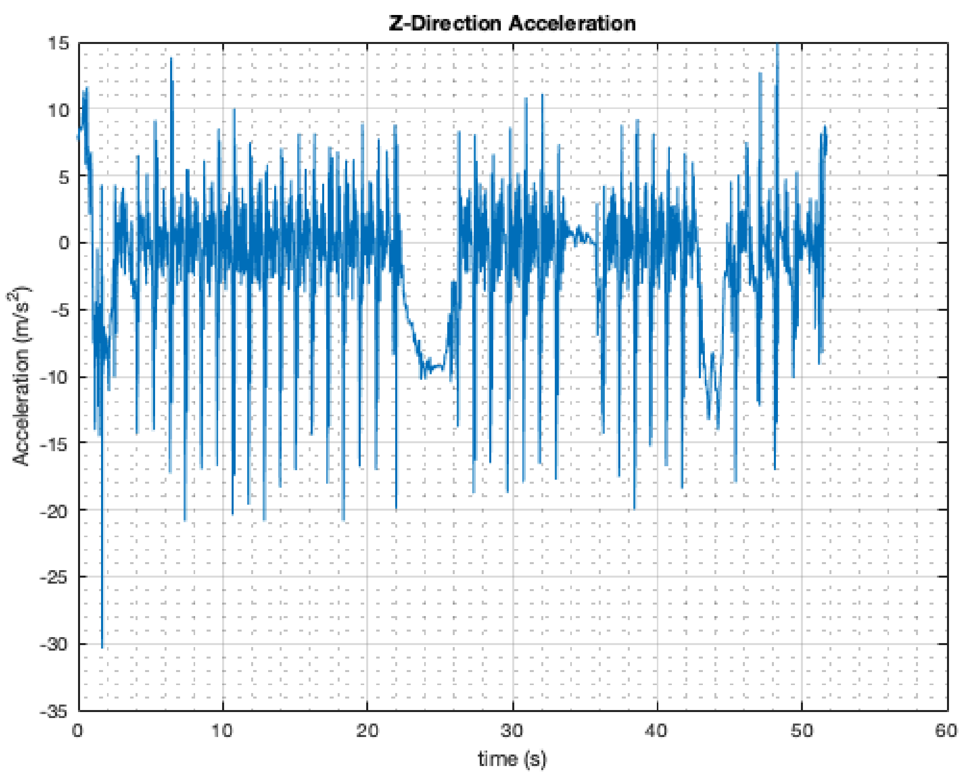
```
count_hf = 0;
thresh_hf = (.5) * max(filt_hf);
i = 1;
```

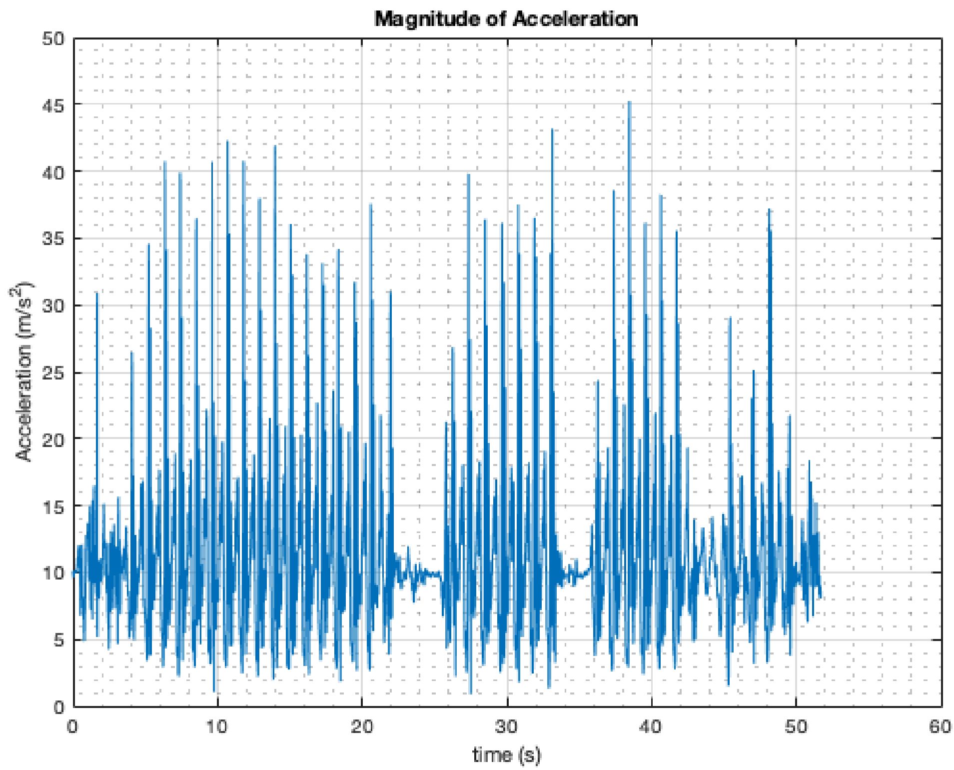
```
while i <= length(filt_hf)
    if filt_hf(i) > thresh_hf
       count_hf = count_hf + 1;
       while filt_hf(i) > thresh_hf
           i = i + 10;
           if i > length(filt_hf)
               break
           end
       end
    end
    i = i + 1;
end

% subtract out jostles from original step count to get final count
msg = "New Filter: You must have taken " + (count - count_hf) + "
 steps!";
disp(msg)
```

Original Filter: You must have taken 76 steps!
New Filter: You must have taken 68 steps!

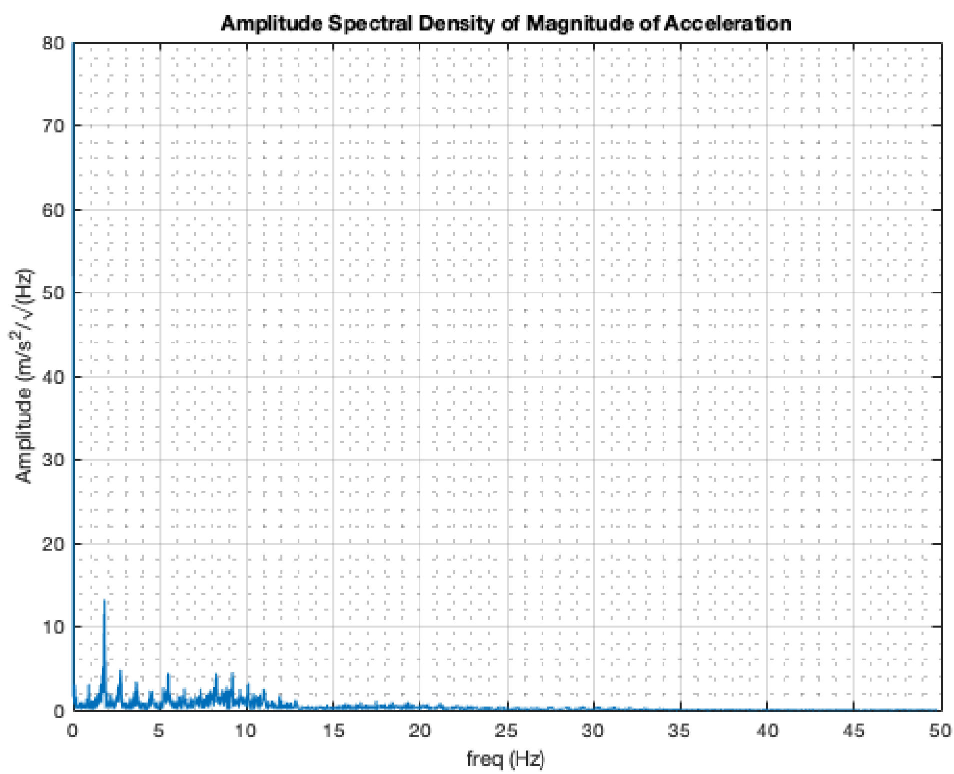



Y-Direction Acceleration
Acceleration (m/s^2)
time (s)
20
10
0
-10
-20
-30
-40
-50
0
10
20
30
40
50
60

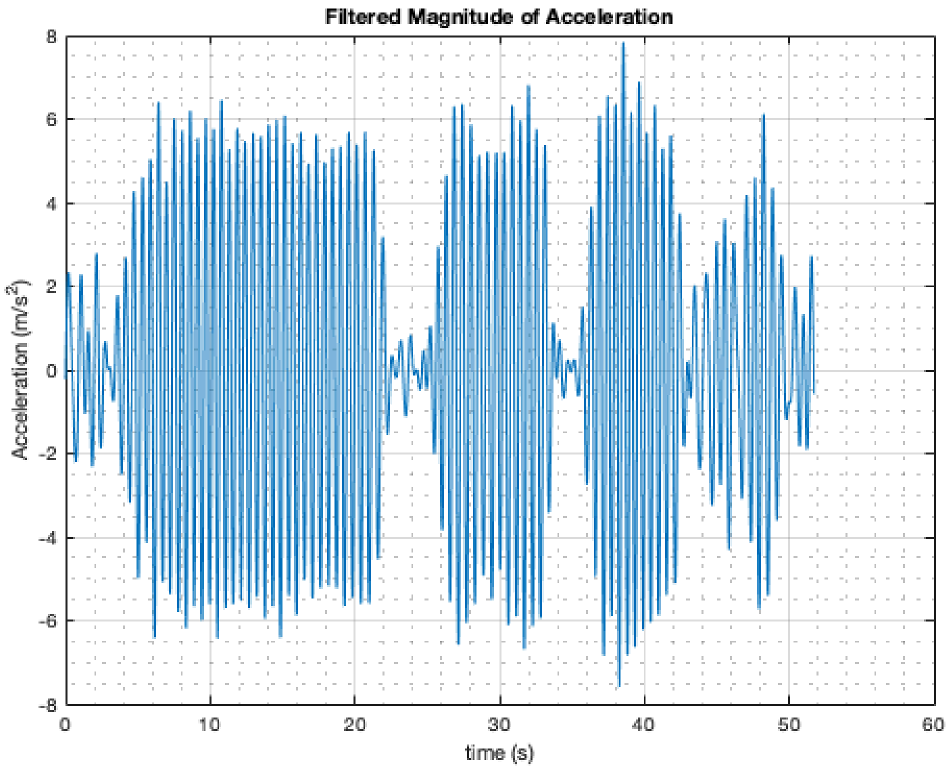

Z-Direction Acceleration
Acceleration (m/s^2)
time (s)
15
10
5
0
-5
-10
-15
-20
-25
-30
-35
0
10
20
30
40
50
60



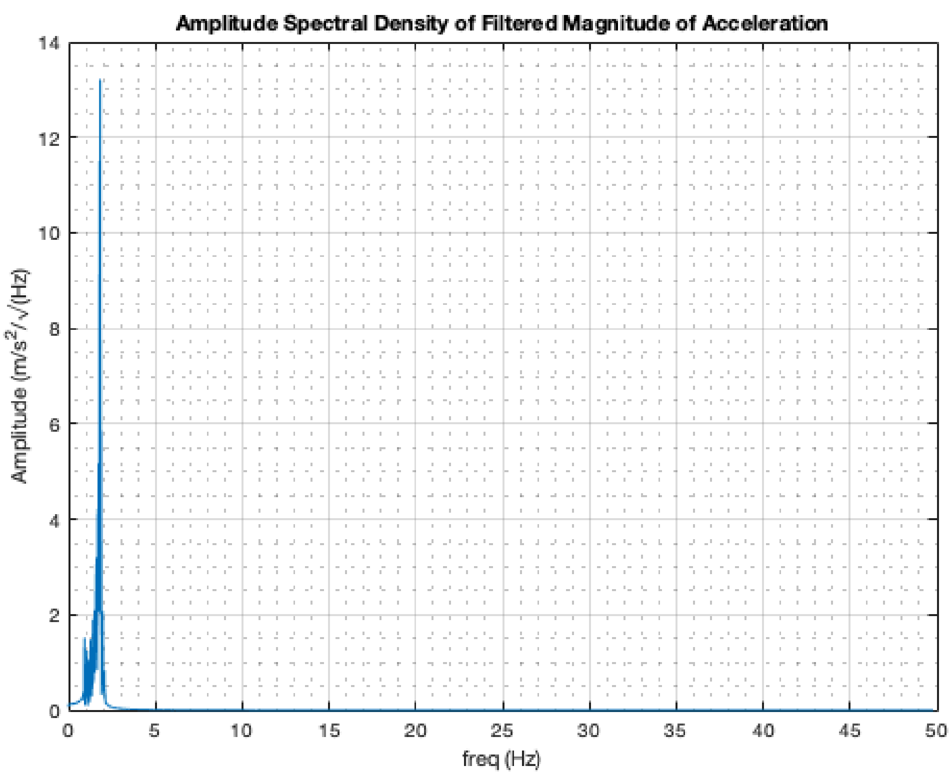
Magnitude of Acceleration
Acceleration (m/s^2)
time (s)
0
10
20
30
40
50
60
5
15
25
35
45



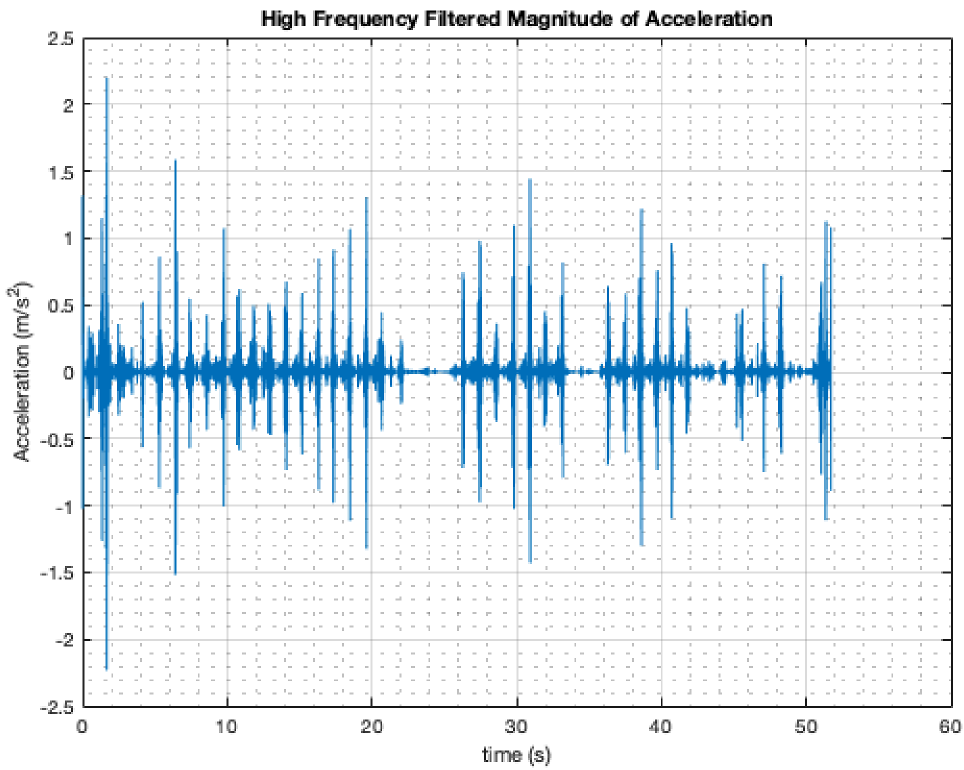
Amplitude Spectral Density of Magnitude of Acceleration
Amplitude (m/s^2/√(Hz)
freq (Hz)
0
10
20
30
40
50
60
70
80
5
15
25
35
45

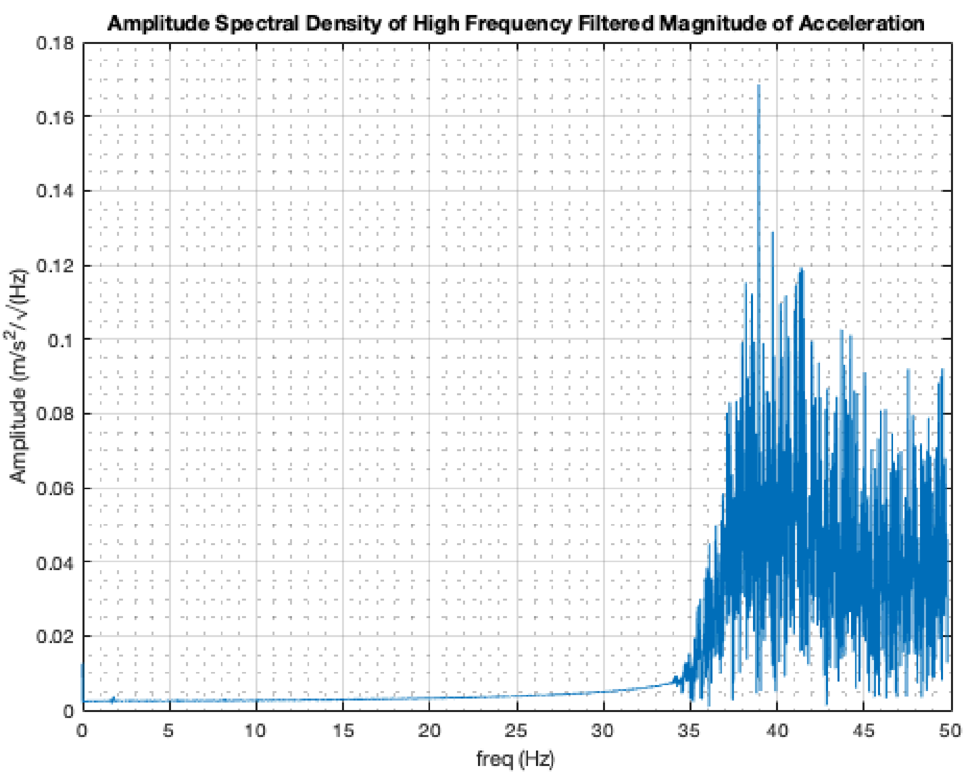


Filtered Magnitude of Acceleration
Acceleration (m/s²)
time (s)
8
6
4
2
0
-2
-4
-6
-8
0
10
20
30
40
50
60



Amplitude Spectral Density of Filtered Magnitude of Acceleration
Amplitude (m/s²/√(Hz)
freq (Hz)
14
12
10
8
6
4
2
0
0
5
10
15
20
25
30
35
40
45
50


High Frequency Filtered Magnitude of Acceleration
Acceleration (m/s^2)
time (s)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
10
20
30
40
50
60


Amplitude Spectral Density of High Frequency Filtered Magnitude of Acceleration
Amplitude (m/s^2/√(Hz)
freq (Hz)
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
0
5
10
15
20
25
30
35
40
45
50

Published with MATLAB® R2020b