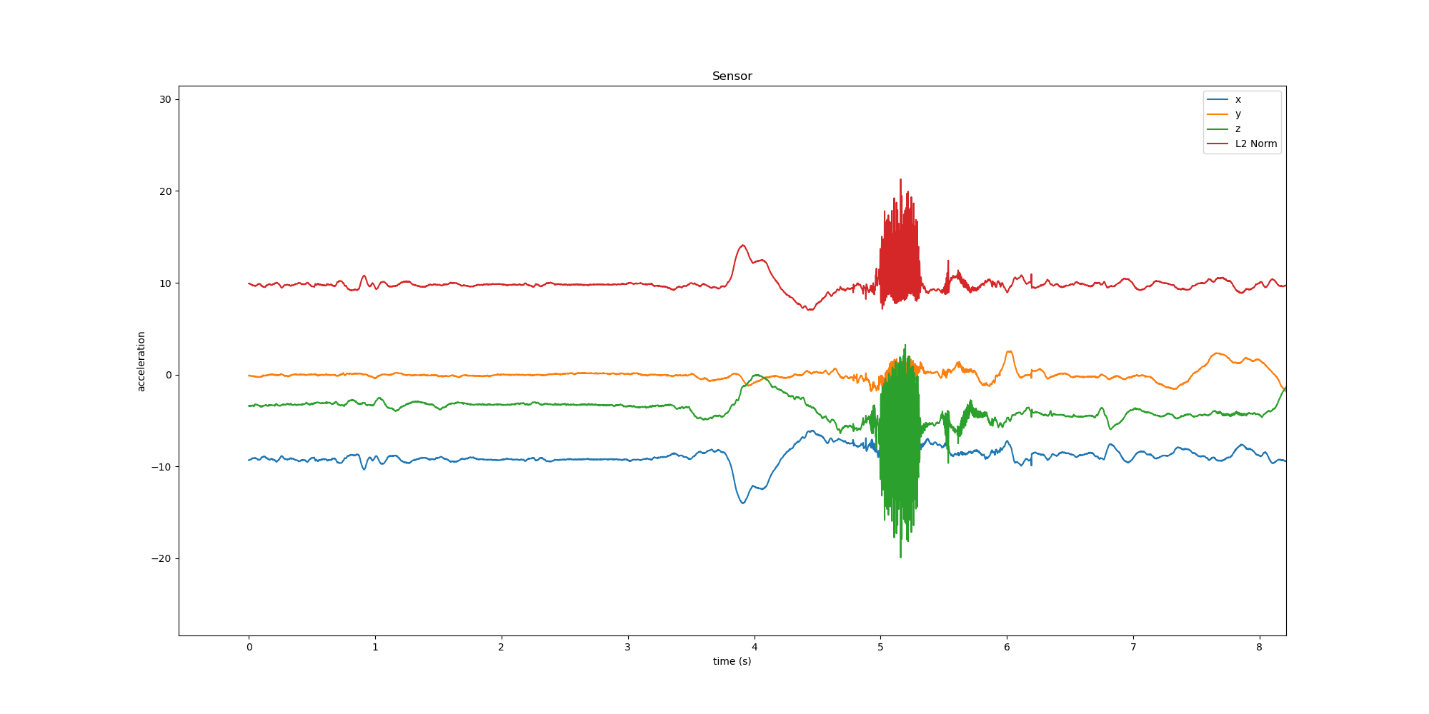
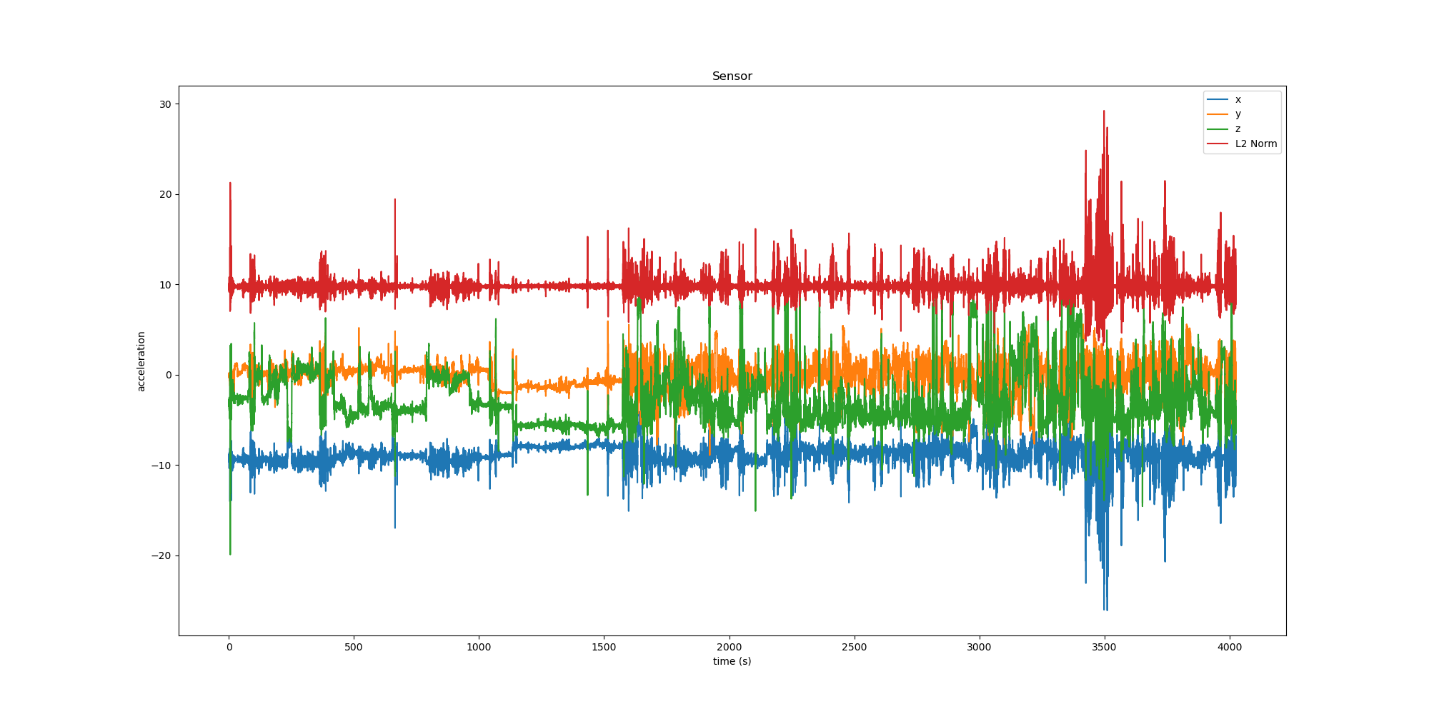
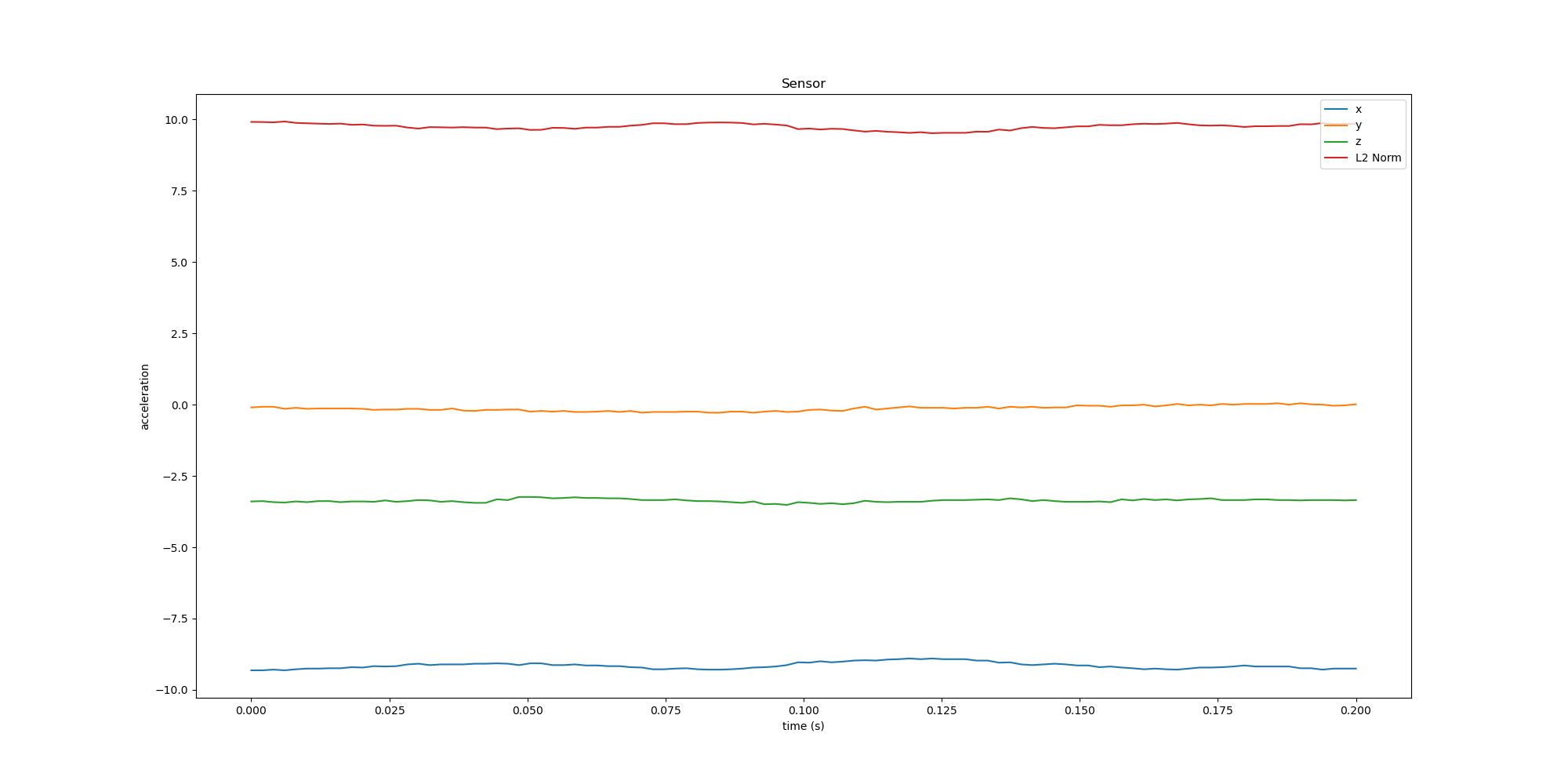
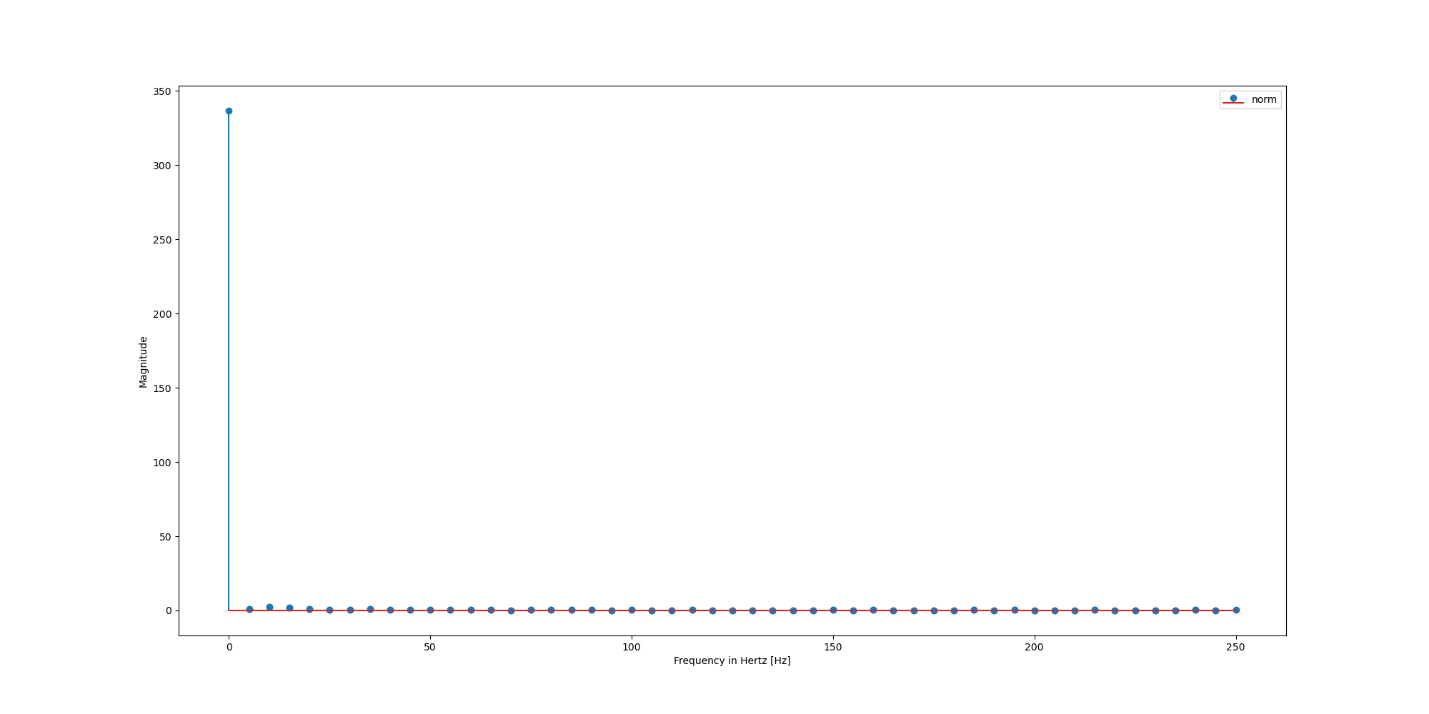
Mikhail Mayers

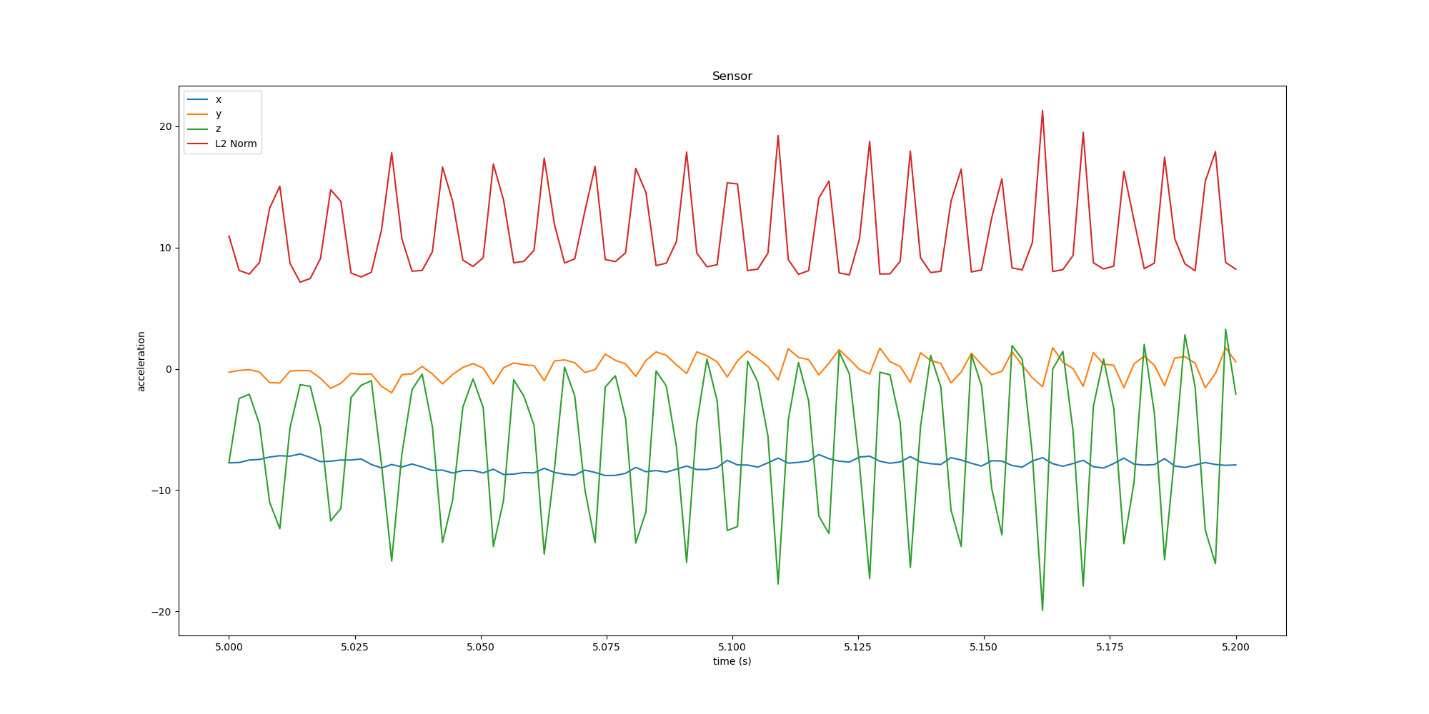
EE523

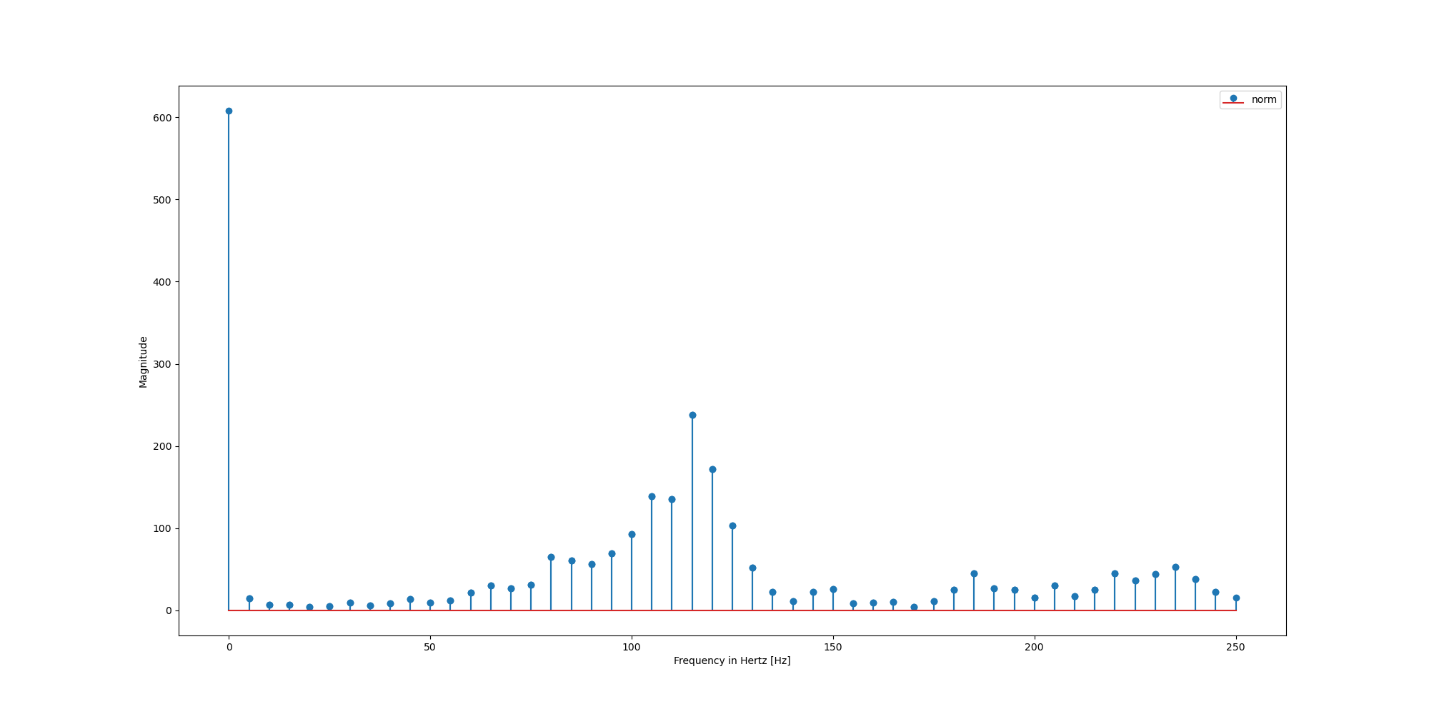
Hw2

Starting out, I knew I would need to determine a region in which there is a high probability that we have a cough or not. Though I tried to implement what we learned in class last Wednesday, I was not successful. I did however come up with a plan based off an aside from class. You mentioned for one example, that we had a cough that was large and easy to see, and then immediately afterwards, was a small cough. I realized that the answer was not in the time domain, but in the frequency domain. 

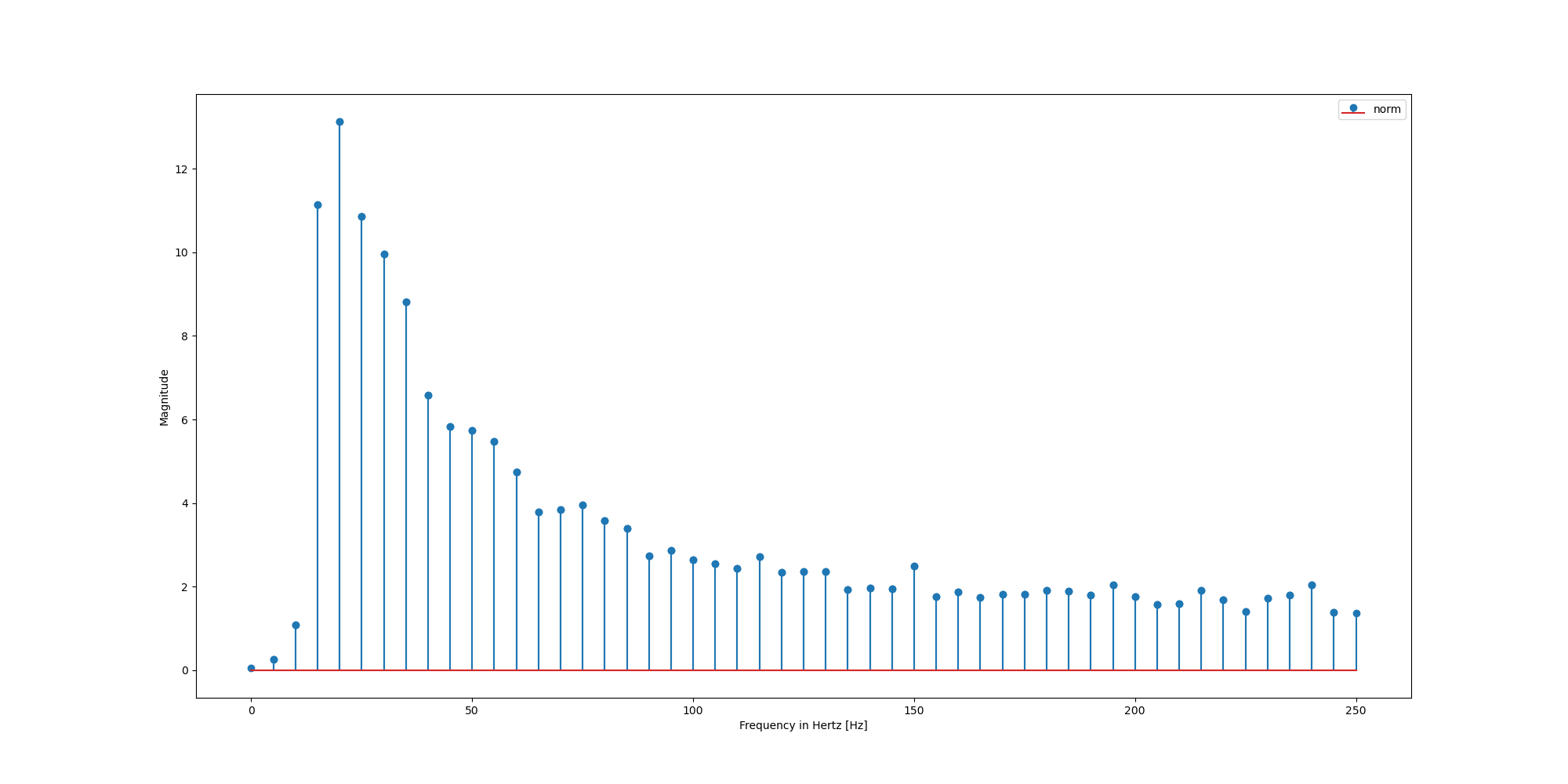
From here I decided in order to accurately detect coughs, I would need to break the recording into small time segments and look for a cough within that time instance rather than the whole sequence. I arbitrarily chose .2 seconds (100 samples) as my time window. I know that coughs usually last less than 1 second, so any number less than one second might be good, and I would rather make something more sensitive to begin with. And from that window I would look in the frequency domain to see what is going on. 

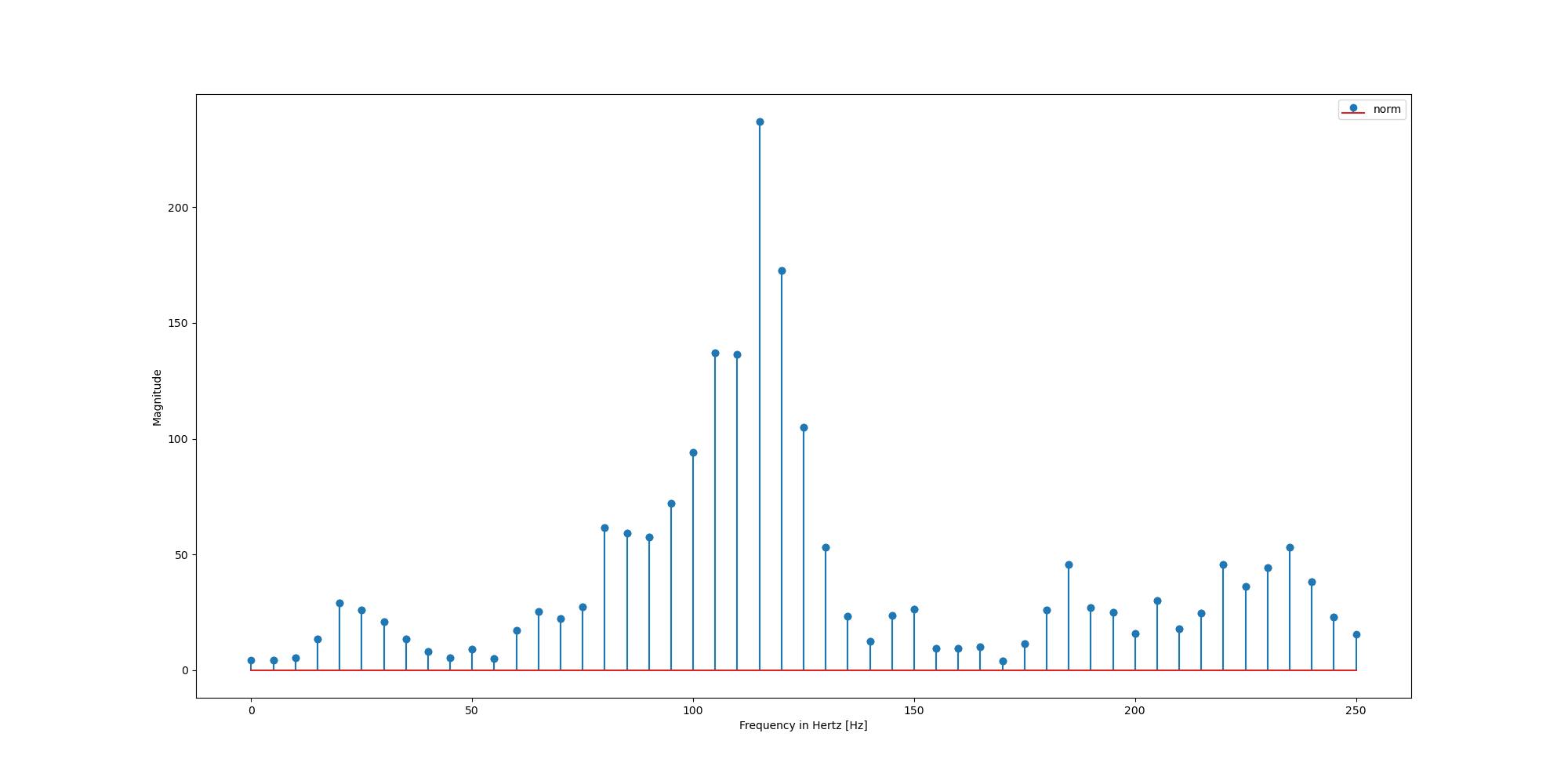


From looking at the frequency I have determined that the noise is most likely gaussian, because it is very high at the center and then quickly tapers off. I know that I will need to filter this, but I still want to see what the frequency domain looks like during a cough. Because I know about the cough at ~5 seconds, I chose it to be my initial test. 

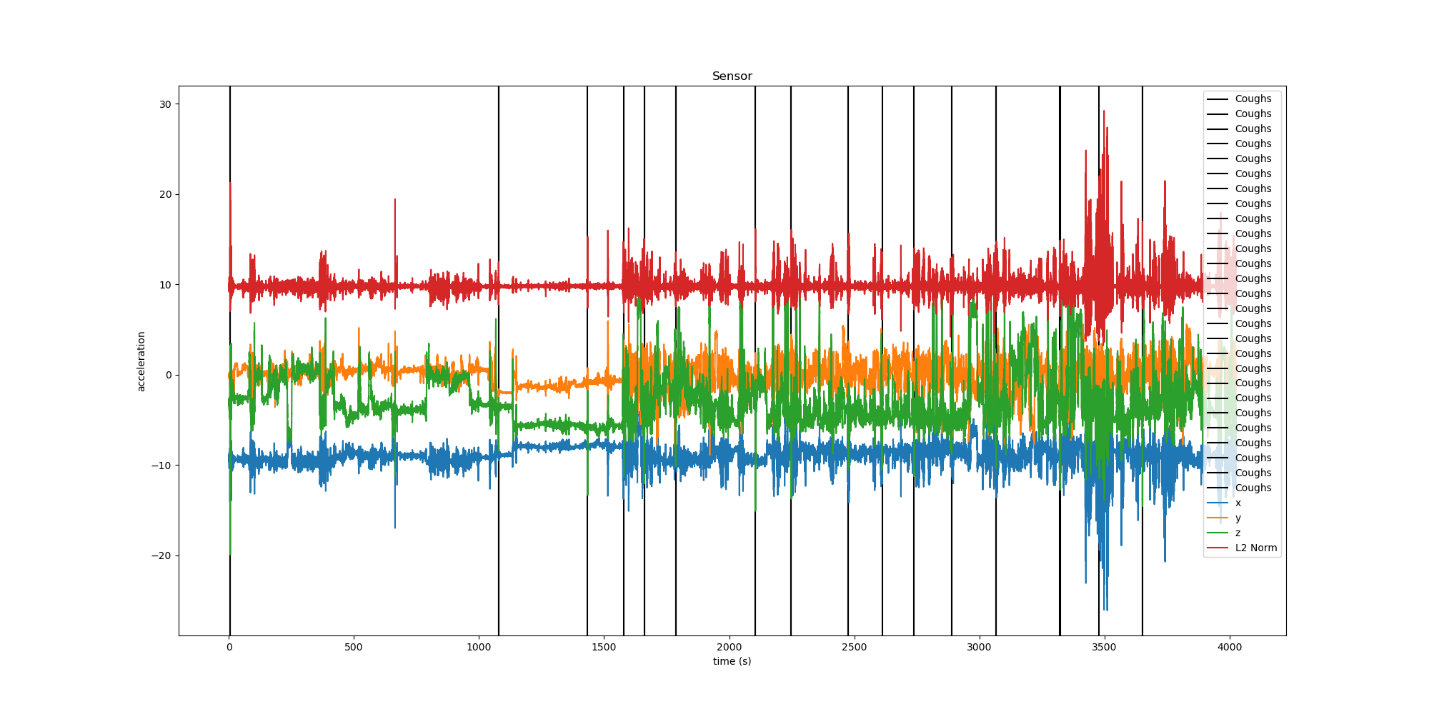


And there is a clear difference, even without removing noise. So, I try the same thing except I filter the data. For this homework, I will ignore trying to make this fast, focus on accuracy and choose a Butterworth filter. With the filter implemented, the difference becomes more apparent.

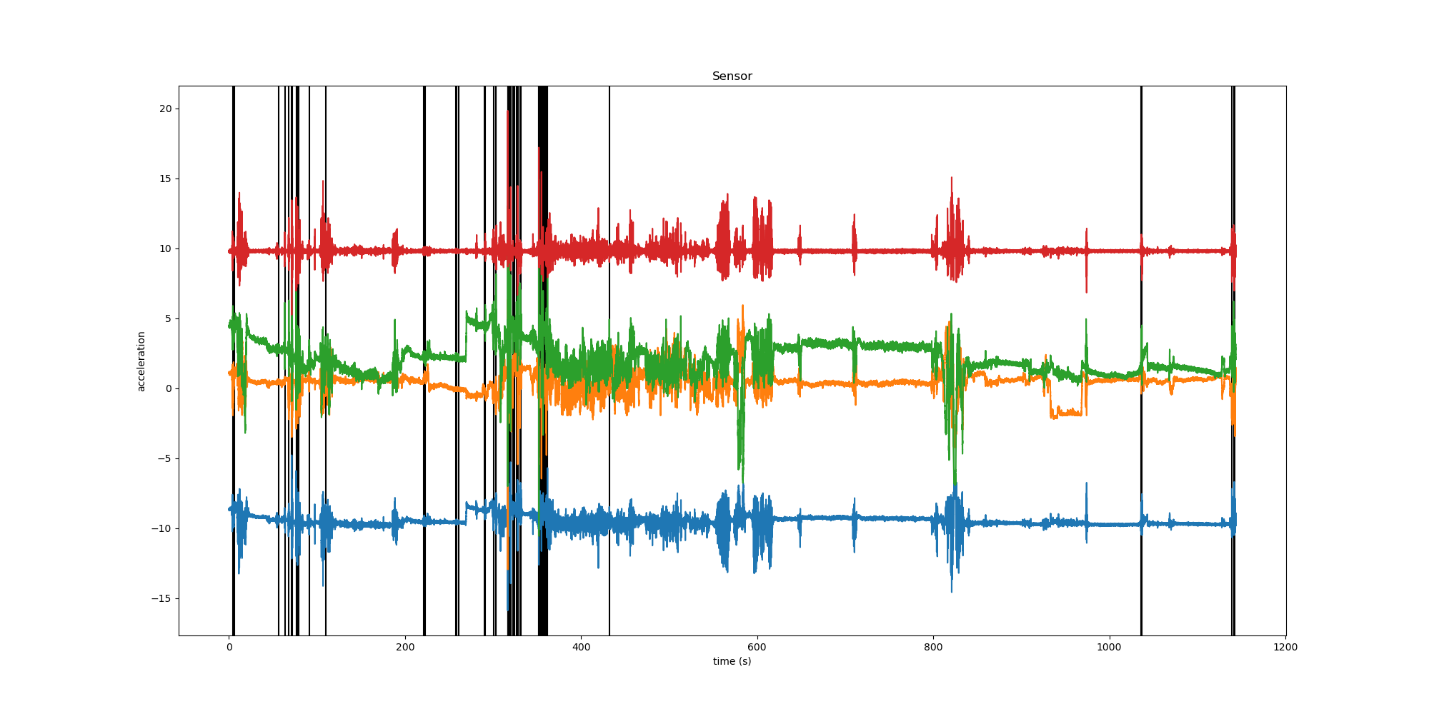


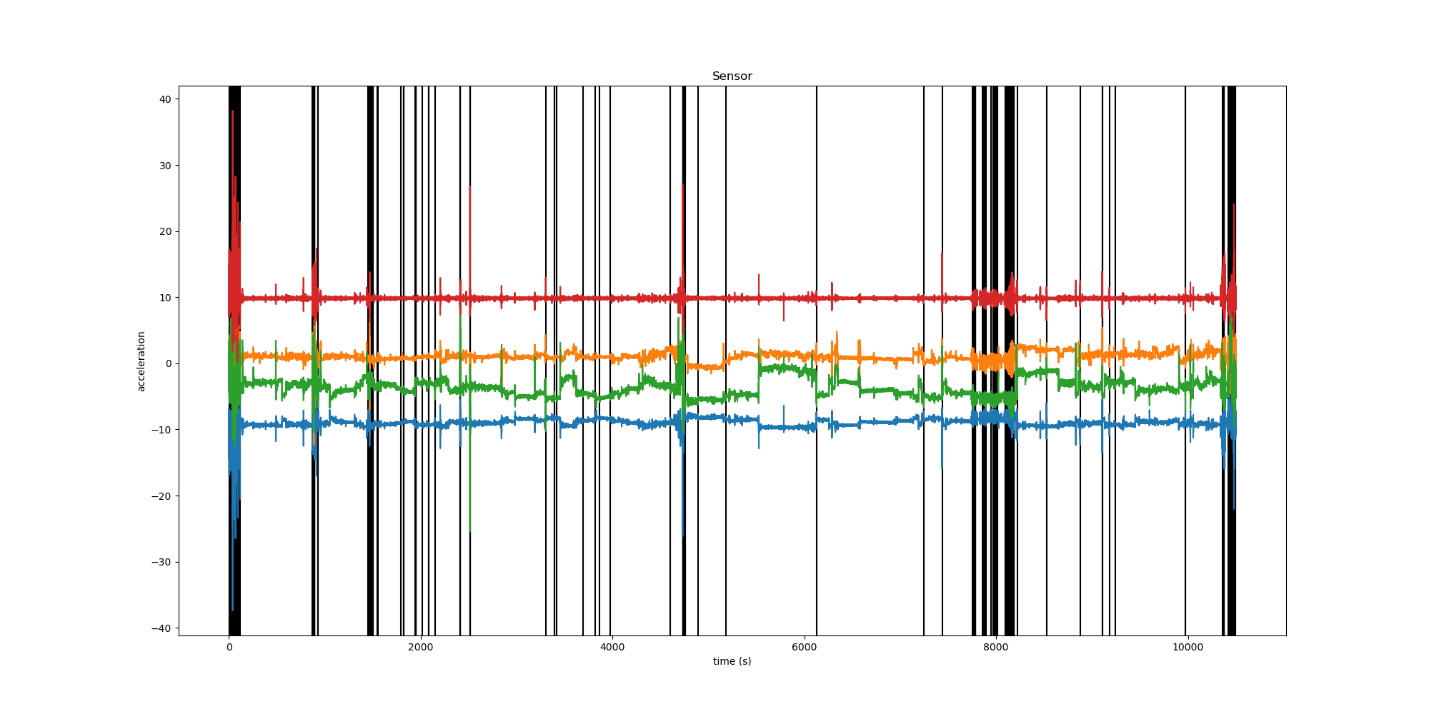


Because I know the times of the cough, I make the window for detection ~1 second with the annotated points provided as the mid points (.5 seconds on each side). I figured this would be a good time interval to look at. My justification is that, if I am more off than 1 second, then I must be way too off for my method to be redeemable. I am focusing on accuracy with as few guesses as possible. From here, I set a threshold that if my frequencies are above a threshold, then that would be a cough prediction. So, I tested my algorithm over the entire data set with my threshold being at magnitude 15.



However, these settings did not work for the second or third cough data sets. Upon looking at the data, the coughs were had a different threshold. I had to lower my threshold down to 5 and change the cutoff frequency of my filter. Though the accuracy was not as good as I had hoped for and my algorithm program predicted more often than before.





I thought this was a good time to quantify my performance. That being said, I did a similar technique to last time. I used my windows of detection as a bin, and then I counted how many times my predictions landed in the bin. If I hit a bin more than once, I only counted it once, and the other predictions were overzealous predictions. So, using the Sensitivity, and specificity equations, tp/ (tp + fn) and 1-(fp/(fp+tn)), I needed to figure out my false positives, true positives, false negatives, and true negatives. True Positives were all the successful times I hit a bin at least once, without counting multiples more than once. False negatives were all the points I missed and did not predict correctly. True negatives were all the points I could predict outside of the windows of detection. Finally, my false positives were all the times I predicted outside of the windows of detection. Plugging my numbers into the equations, I got the following.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Sensitivity | Specificity | Predictions | Total data points |
| R001(cutoff 15, thresh 15) | .85 | 0.9999950306534144 | 27 | 20120 |
| R002(cutoff 1, thresh 5) | .75 | 0.9999177375372369 | 62 | 5710 |
| R003(cutoff 1, thresh 5) | .75 | 0.999932747416634 | 368 | 52490 |

I think it could be feasible to use an accelerometer on the sternum to detect coughs, though I think I need more tools and knowledge to do it. A better way to predict coughs would make this better, because with my method, I could at least find most of the points, though I over predicted a lot. From this homework, I did learn to check out the frequency domain, though I did try to predict using PSD and the frequency domain with a log scale, but I did not gain any useful information (that I could see) for my predictions. I would like to lean how to do this in a better way and learn how I can use what we learned in class to make my predictions better. I think I am on the right track, but there are parts that I am missing to make it a reality. So, my new questions are: how can I use probabilities in the equations, and what other tools will make this work better?