plt.plot(f[:300], P[:300]) 🡨This stopped working for me for some reason so I moved to what I was familiar with – MATLAB ☺

What I did:

1. Run LFP\_Filtering.py (on cluster)
   1. Selected EMG channels to [24,25]
2. Run LFP\_Trial\_Parsing.py
   1. Digital input channels were:
      1. 0: 90 Trials (30 NaCL (15 first/15 last), 60 LiCl)
      2. 1: 30 Trials (30 Water)
   2. Durations:
      1. Pre-Stimulus: 2000ms
      2. Post-Stimulus: 5000ms
   3. Delete Raw LFP data:
      1. Yes
3. [Fast Fourier transform](https://en.wikipedia.org/wiki/Fast_Fourier_transform) (FFT):
   1. Use this code:

#Import necessary tools

import numpy as np

import tables

import matplotlib.pyplot as plt

from scipy import signal

from scipy.signal import periodogram

#Open up hf5 file

hf5 = tables.open\_file('NM46\_CTA\_training\_160929\_130610\_repacked.h5', 'r+')

#Import Data for Water and LiCl trials (Dig\_in 0) and NaCl (Dig\_in\_1)

data = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:]

W1\_data = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,:15,:]

LiCl\_data = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,15:75,:]

W2\_data = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,75:,:]

NaCl\_data = hf5.root.Parsed\_LFP.dig\_in\_1\_LFPs[:,:30,:]

NaCl\_data1 = hf5.root.Parsed\_LFP.dig\_in\_1\_LFPs[:,:15,:]

NaCl\_data2 = hf5.root.Parsed\_LFP.dig\_in\_1\_LFPs[:,15:30,:]

#Whole Trial Parsing

W1\_data = hf5.root.Parsed\_LFP.dig\_in\_1\_LFPs[:,:15,:]

W2\_data = hf5.root.Parsed\_LFP.dig\_in\_1\_LFPs[:,15:30,:]

NaCl\_data1 = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,:15,:]

NaCl\_data2 = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,75:,:]

LiCl\_data = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,15:75,:]

#LiCl Parse

LiCl\_data\_1\_4 = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,15:30,:]

LiCl\_data\_2\_4 = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,30:45,:]

LiCl\_data\_3\_4 = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,45:60,:]

LiCl\_data\_4\_4 = hf5.root.Parsed\_LFP.dig\_in\_0\_LFPs[:,60:75,:]

#Get means based on electrodes (0 = num\_electrodes, 1= num\_trials, 2 =durations[0] + durations[1])

W1\_data = np.mean(W1\_data, axis = (0))

W2\_data = np.mean(W2\_data, axis = (0))

NaCl\_data1 = np.mean(NaCl\_data1, axis = (0))

NaCl\_data2 = np.mean(NaCl\_data2, axis = (0))

LiCl\_data = np.mean(LiCl\_data, axis = (0))

LiCl\_data\_1\_4 = np.mean(LiCl\_data\_1\_4, axis = (0))

LiCl\_data\_2\_4 = np.mean(LiCl\_data\_2\_4, axis = (0))

LiCl\_data\_3\_4 = np.mean(LiCl\_data\_3\_4, axis = (0))

LiCl\_data\_4\_4 = np.mean(LiCl\_data\_4\_4, axis = (0))

#FFTs

f, P = periodogram(data, fs = 1000.0, return\_onesided=True)

f\_W1, P\_Water1 = periodogram(W1\_data, fs = 1000.0, return\_onesided=True)

f\_W2, P\_Water2 = periodogram(W2\_data, fs = 1000.0, return\_onesided=True)

f\_NaCl1, P\_NaCl1 = periodogram(NaCl\_data1, fs = 1000.0, return\_onesided=True)

f\_NaCl2, P\_NaCl2 = periodogram(NaCl\_data2, fs = 1000.0, return\_onesided=True)

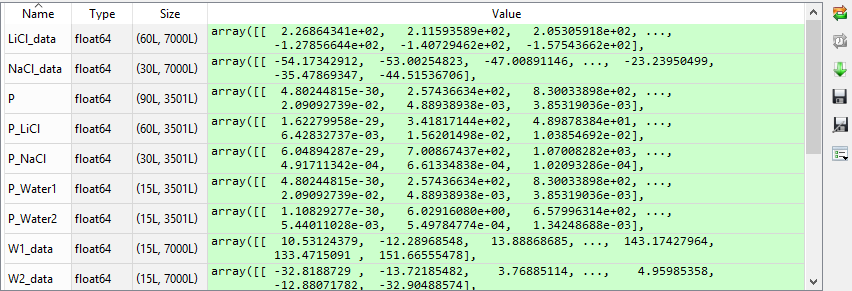
f\_LiCl, P\_LiCl = periodogram(LiCl\_data, fs = 1000.0, return\_onesided=True)

f\_LiCl14, P\_LiCl14 = periodogram(LiCl\_data\_1\_4, fs = 1000.0, return\_onesided=True)

f\_LiCl24, P\_LiCl24 = periodogram(LiCl\_data\_2\_4, fs = 1000.0, return\_onesided=True)

f\_LiCl34, P\_LiCl34 = periodogram(LiCl\_data\_3\_4, fs = 1000.0, return\_onesided=True)

f\_LiCl44, P\_LiCl44 = periodogram(LiCl\_data\_4\_4, fs = 1000.0, return\_onesided=True)



1. Save the data
   1. Click on the “Save data as…” button in Python
   2. Save data with whatever name you want with “\*.mat” extension
2. Plotting
   1. Open MATLAB and use this code:

%load data

Change this to whatever you saved your data as.

load('Practice.mat')

%loop through PSD data to create averages across trials

for i=1:length(P\_water1(1,:)),

P\_water1(16,i) = mean(P\_water1(1:length(P\_water1(:,1)),i));

end;

for i=1:length(P\_NaCl(1,:)),

P\_NaCl(31,i) = mean(P\_NaCl(1:length(P\_NaCl(:,1)),i));

end;

for i=1:length(P\_LiCl(1,:)),

P\_LiCl(61,i) = mean(P\_LiCl(1:length(P\_LiCl(:,1)),i));

end;

for i=1:length(P\_water2(1,:)),

P\_water2(16,i) = mean(P\_water2(1:length(P\_water2(:,1)),i));

end;

Change this based on trials per tastes.

%Plot

plot(f\_W1,P\_water1(16,:));

hold on;

plot(f\_NaCl,P\_NaCl(31,:));

hold on;

plot(f\_LiCl,P\_LiCl(61,:));

hold on;

plot(f\_W2,P\_water2(16,:));

hold on;

yW1max = max((P\_water1(16,:)));

yNamax = max((P\_NaCl(31,:)));

yLimax = max((P\_LiCl(61,:)));

yW2max = max((P\_water2(16,:)));

ylim = max([yW1max,yNamax,yLimax,yW2max]);

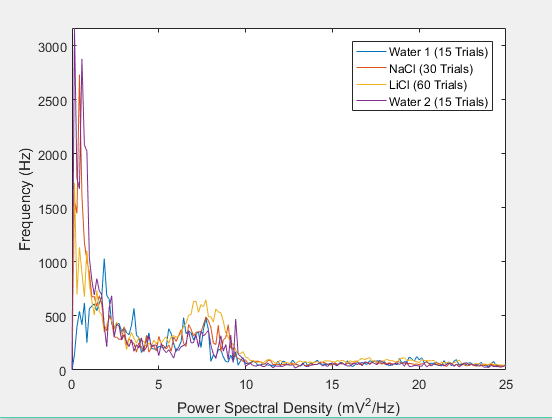
axis([0 25 -1 ylim]);

xlabel('Power Spectral Density (mV^{2}/Hz)');

ylabel('Frequency (Hz)');

legend('Water 1 (15 Trials)','NaCl (30 Trials)','LiCl (60 Trials)','Water 2 (15 Trials)','Location','northeast');

1. Output



1. Output 2 (Separated NaCl trials)

%load data

load('Updated\_NaCl1\_2.mat')

%loop through PSD data to create averages across trials

for i=1:length(P\_Water1(1,:)),

P\_Water1(16,i) = mean(P\_Water1(1:length(P\_Water1(:,1)),i));

end;

for i=1:length(P\_NaCl1(1,:)),

P\_NaCl1(16,i) = mean(P\_NaCl1(1:length(P\_NaCl1(:,1)),i));

end;

for i=1:length(P\_NaCl2(1,:)),

P\_NaCl2(16,i) = mean(P\_NaCl2(1:length(P\_NaCl2(:,1)),i));

end;

for i=1:length(P\_LiCl(1,:)),

P\_LiCl(61,i) = mean(P\_LiCl(1:length(P\_LiCl(:,1)),i));

end;

for i=1:length(P\_Water2(1,:)),

P\_Water2(16,i) = mean(P\_Water2(1:length(P\_Water2(:,1)),i));

end;

%Plot

figure(200);

plot(f\_W1,P\_Water1(16,:));

hold on;

plot(f\_NaCl1,P\_NaCl1(16,:));

hold on;

plot(f\_NaCl2,P\_NaCl2(16,:));

hold on;

plot(f\_LiCl,P\_LiCl(61,:));

hold on;

plot(f\_W2,P\_Water2(16,:));

hold on;

yW1max = max((P\_Water1(16,:)));

yNa1max = max((P\_NaCl1(16,:)));

yNa2max = max((P\_NaCl2(16,:)));

yLimax = max((P\_LiCl(61,:)));

yW2max = max((P\_Water2(16,:)));

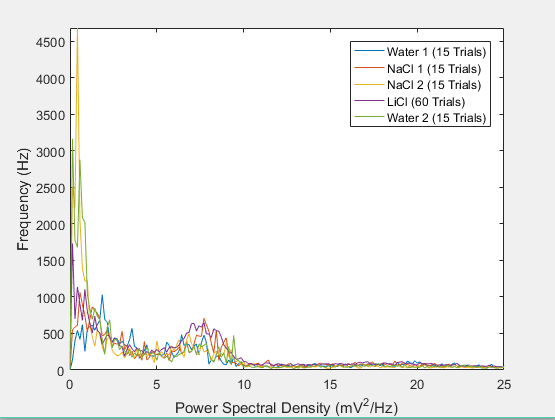
ylim = max([yW1max,yNa1max,yNa2max,yLimax,yW2max]);

axis([0 25 -1 ylim]);

xlabel('Power Spectral Density (mV^{2}/Hz)');

ylabel('Frequency (Hz)');

legend('Water 1 (15 Trials)','NaCl 1 (15 Trials)','NaCl 2 (15 Trials)','LiCl (60 Trials)','Water 2 (15 Trials)','Location','northeast');



1. Output 3 (Single NaCl trials)

figure(200);

hold on;

plot(f\_NaCl1,P\_NaCl1(16,:));

hold on;

plot(f\_NaCl2,P\_NaCl2(16,:));

hold on;

yNa1max = max((P\_NaCl1(16,:)));

yNa2max = max((P\_NaCl2(16,:)));

ylim = max([yNa1max,yNa2max]);

axis([0 25 -1 ylim]);

xlabel('Power Spectral Density (mV^{2}/Hz)');

ylabel('Frequency (Hz)');

legend('NaCl 1 (15 Trials)','NaCl 2 (15 Trials)' ,'Location','northeast');

