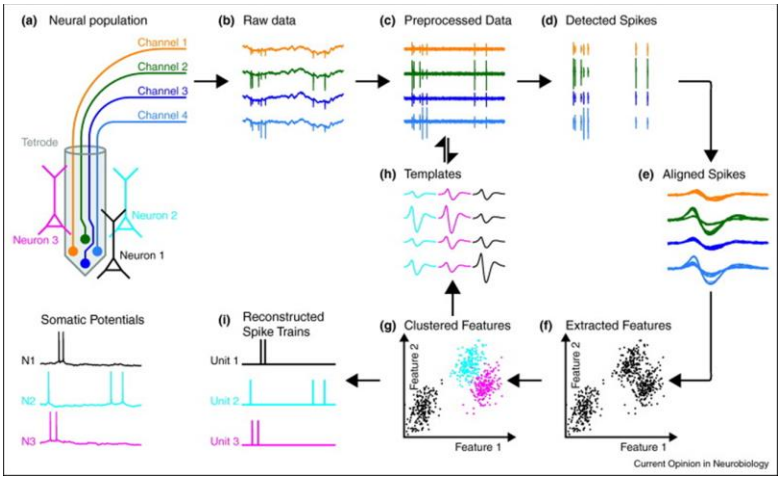
Neural Implants and Interfaces

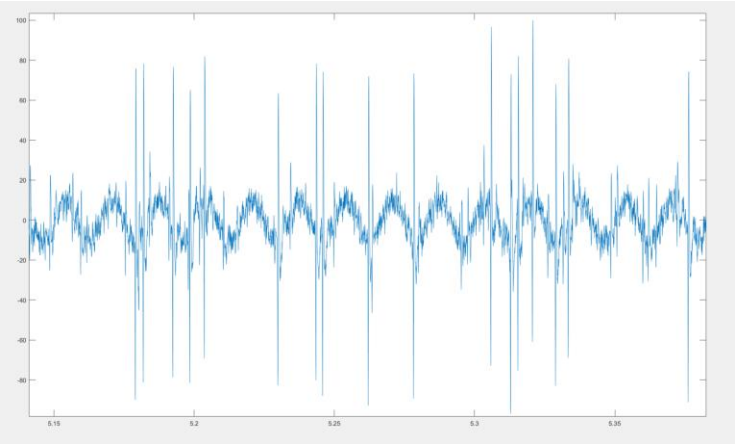
Homework 3

Fig 1.



In this homework you are given HW3_datafile.mat. You can load the file using the "load" Matlab command. The file contains "Data" and "fs". "fs" is the sampling rate of the data.

This file contains ~25s of data recorded from one electrode implanted in the brain. The data file is corrupted with 60Hz noise from the AC line. Here's a short 200ms sample from the file as an example.



Your job is to create a template recognition system that would classify spikes. You will split the file into a 5s training set and use the rest as the test set. You will need to classify the spikes from the test set as belonging to a particular neuron. i.e. you need to get to part "i" in the top part of the figure.

For each of the steps provide a Matlab script and a 1s snippet of the processed data that demonstrates that you accurately accomplished the task (unless it asks for some other plot).

Do this for the 5s test set.

1. Band-pass filter the data between 300Hz and 3kHz to eliminate the noise (Fig1 (c)). **Plot 1s snippet showing the filtered data.**

2. Use NEO to detect spikes (Fig1 (d)). **Plot the 1s snippet of NEO and the corresponding original data with the spikes identified on the plot.**

3. As you detect spikes, align them at the maximum and extract as a window of 1ms before the max and 2ms after (Fig1 (e)). **Plot these on top of each other and show that plot.**

4. After you collect all of the spikes, create PCA to extract (Fig1 (f)) the principle components (Fig1. (f)). **Plot the first two principle components as in Fig1 f.**

5. Cluster the features using k-means algorithm (Fig1 (g)). If k-means does not result in a "good" classification of the features, use a "manual" method.
   Plot and color-code as in Fig1 g.

6. Create time plots for each neuron in the test set. You have a choice on how to do this (do only one):
   a. Use a template based algorithm by creating an averaged template for each neuron's spike in the training set and then running a window along the FILTERED training set looking for the spikes with MSE approach.
   b. Use a kNN approach by doing a PCA on each unknown spike in the training set after you detect them through steps 1-3.

   Plot 1s snippets aligned in time showing the original dataset with the classified detection of spikes for each neuron in a separate row (i.e. plot a "1" if that neuron's spike is detected or 0 if not).