**Scripts needed for data analysis:**

**Run\_Batches** allows you to run all the data for the specified dates in one run, instead of running each date individually. It’s very convenient and this has become the go to script for analysis.

**Format\_Data\_v2** takes the Intan files (.rhd) and formats for analysis, and it will save the structure into a .mat file with the same name as the rhd filename. It generates one .mat file per whisker

Inputs:

* Sampling rate
* Recording device

Outputs:

* NeuralData structure that contains:
  + Filename
  + Channel Impedance
  + Whisker ID
  + Channel number
  + LFP
  + Gamma
  + Time
  + Sampling rate
  + Pulse signal

It is optimized to run all files in one folder. Once it had been run once, it does not need to be run again.

**Func\_Data\_Preprocessing\_v13** will upload the .mat files created in Format\_Data and will preprocess the data needed for analysis.

Inputs:

* Date
* Device, DISC or tetrode
* Window duration: time in seconds, it's the time before and after the pulse. A .200 window would have analysis span from 0 to 200 ms.
* Target\_trial\_num, is the number of trials that will be used
* Model\_flag, to indicate if model will be run (1) or not (0)
* CAR\_Flag, indicates if common average referencing by subtracting mean of all channels will be performed (1) or not (0).
* Num\_ch\_RV, is a vector with the number of channels that will be used to generate the directional curve and resultant vector
* Num\_ch is the number of channels that will be used for the analysis. This is a predetermined DISC electrode configuration. Values accepted:
  + 128 (16 rows x 8 columns)
  + 64 (16 x 4)
  + 88 (11 x 8)
  + 44 (11 x 4)
  + 24 (3 x 8)
  + 12 (3 x 4)
  + 8 (1 x 8)
  + 4 (1 x 4)
  + 1 (1 x 1) . This is the best channel with highest amplitude.

Tetrode device only accepts 4 and 1 electrodes

Parameters that can be modified in text:

* Binsize - degree difference between interpolated angles for directional curves
* Gammaflag, to indicate if it’s gamma (1) or LFP (0) bands being analyzed
* Saveflag is used to indicate the program if the figures that are being generated to demonstrate the confusion matrix should be saved (1) or not (0)
* Macroflag will tell the script whether to perform DISC macro modeling (1)  
   or not (0)
* Kept\_stdev is the number of standard deviations away from the mean will be used for denoising. If kept\_stdev=3, all the data 3 standard deviations from the mean will be kept. Everything else will be thrown away. This is not currently active in this version of the code, as the denoised sections are commented
* Percentage\_data\_testing is the percentage of data that will be used for testing. 100-percentage\_data\_testing is the percentage that will be used for triaing.
* Target\_variance is the percentage of the variance that wants to be explained by the number of principal components chosen. Therefore, the Func\_LDA will chose as many principal components as it takes to get to the target\_variance percentage
* If plotting RV or polar plots:
  + Type, LFP or Gamma, this should match the value given by GammaFlag
  + Plot\_type

Outputs:

* Model\_summary, structure with the accuracy, snr, rms, amplitude for each whisker and device. You get 10 repetitions per whisker, as this is equivalent to the number of crossvalidation used in the model
* Data\_summary, structure with the max snr, max amplitude, mean rms noise, max rms signal for each whisker and device. Individual directional curves are also saved here.

Things done in this script:

* For all channels:
  + Separate data into trials, and for each trial calculate:
    - amplitude
    - SNR
    - RMS noise and signal is implemented but commented. Uncoment if needed
  + Average waveform of all trials, and calculate the amplitude, and SNR
* For channels in channels\_interest.RV:
  + Calculate a directional curve using the average waveform
  + Calculate a directional curve for each trial
* Generate feature matrix for each trial by saving gamma snippets (voltage vs time) for all channels in channels\_interest.model. The neural activity voltage is acquired every .5 ms and it goes from 0 to +windowduration.
  + The directional curve for each trial can be added to this feature matrix
  + The resultant vector of the addition of individual directional curves is sometimes also added in lieu of the directional curve
* If model\_flag is on, it will call the func\_model\_all\_closest\_furthest\_whiskers\_v2 to:
  + Divide the data into training and testing data
  + Do PCA for dimensionality reduction
  + Run LDA using 5 and 9 classes
  + Calculates model accuracy

Channels being analyzed were previously chosen (see DISC\_KeyFacts). If it needs to be changed, modify the channels\_interest structure

**Func\_GetSnippets** separates waveform into individual trial snippets by finding the pulse signal start and looking at –windowduration to +windowduration data points. Trial data will be stored in a matrix with each row being a trial, and each column the voltage value at a specific point in time

**Func\_FilterDownsampleLFP\_v2**

**Func\_LDA\_v6** is where the LDA model is done. It splits the data into 10 folds, does PCA for dimensionality reduction, runs LDA using all whiskers, and calculates model accuracy. confusion matrices for each fold was implemented in version 5, but removed in version 6

**Func\_Model\_All\_Closest\_Furthest\_Whiskers\_v2** creates model using Func\_LDA\_v6 using all whiskers, closest and furthest. The closest and furthest implementation is usually commented. Needs the feature matrix as input. Runs the Func\_LDA\_v5 to train the model. It’s written to also find the closest and furthest whiskers ranked by amplitude values. This part is commented, so if it wants to be run it has to be uncommented. It generates confusion matrix points if plot\_flag is on

**ChooseDirectory** changes matlab path based on device type and date to redirect the code where the whisker data is saved.

**Choose\_Channels\_v2.** Based on the date and number of channels, it selects the channels used for model, simulated macro and directional curve.

**Func\_MW** generates feature matrix for microwire

**Func\_MacroDISC** generates feature matrix for macro

**Func\_DISC\_RV** generates feature matrix for DISC device. By default it is written to take the feature matrix from 0 to +windowsduration. However, you can manually uncomment two sections of the code to indicate whether to take the features from +15 to +windowsduration or -windowduration to +windowsduration

**Func\_RV\_v3**. Based on the tuning curves, it converts them to a resultant vector. If using circular statistics, it computes it here. This is done for each trial

**Func\_TuningCurves \_v3** generates tuning curves and smooths the signal. Device angles automatically updated

**read\_Intan\_RHD2000\_file\_nonotch\_automatic**