**Guide for spike analysis**

**Data analysis pipeline**

**Copy data**

1. Copy the data folder from the recording computer to the server.

**Preprocess data for Kilosort**

1. In load\_command\_KiloSort\_A1.m, change the variables experimentName and sessionName. Run the first section
2. In the 2nd section insert the new parameters and run the section
3. In the 3rd section, modify selCh if necessary and run the section
4. Decide based on std and max which trials to keep
5. In the 4th section type in the exclude variable the subtrial numbers to be excluded. Run the section
6. In the 5th section choose whether to keep the suggested gain or use a different one (choose between 10, 20, 50 or 100, so that there are not too many data points exceeding 32768) and run the section
7. Run the 6th section

**Run Kilosort and Phy2 analysis**

1. Type kilosort (after having added its path upon first use ever)
2. Fill in the fields by copying the file name and deleting/autofilling the folder names
3. Number of channels = 16 or 32; Sampling freq = 20000; min firing rate per channel = 0; Threshold = 10 6; Lambda = 10; AUC for splits = 0.9
4. Run the first lines in SpikeDataLoading\_openEphys\_KiloSort\_A1.m : Clear variables (clearvars -except experimentName sessionName), set the ks parameters (ks= get(gcf, 'UserData'); ks.ops.fshigh = 300; ks.ops.Th = [10 6])
5. Press Run in kilosort
6. Open the Anaconda Prompt (windows) or Terminal (Linux) in the kilosort analysis folder
7. Type „activate phy2” (Windows) or „source activate phy2“ (Linux)
8. Type „phy template-gui params.py“.
9. In the SpikeDataLoading\_openEphys\_KiloSort\_A1.m script in Matlab run the 1st, 2nd and 3rd sections
10. Open the PlotPSTHandRaster\_openEphys\_KiloSort\_A1.m script in Matlab, Run the 1st section
11. Run the 2nd section to get the time courses of good units
12. Use the created figures (together with step 13 and 14) to help decide if a cluster is noise (heartbeat? Artefact? Spike count >200?) or if two clusters are double counted (large bin in the middle of the CCG) or should be merged.
13. To confirm clusters containing noise, in Phy, go to CorrelogramView, Set window to 1000 and go through the clusters and detect heartbeat noise (6-8 Hz frequency, appears in all channels) or other types of noise (optoelectrical artefacts). Then reset window to 50 and Set bin to 0.1 (ms)
14. Go through all the good clusters and compare them with other good clusters based on their similarity. Use Matlab figures created at step 12 to decide if clusters are double counted or should be merged.
15. Go through all the Mua clusters and decide if they are noise (heartbeat? spike count < 200?). Otherwise mark them as Mua. Similarly as for good clusters, time courses can be plotted in Matlab by running the 5th section of PlotPSTHandRaster\_openEphys\_KiloSort\_A1.m
16. When done classifying the clusters in Phy: File/Save; close all in Matlab

**Run matlab analysis**

1. Run the 1st section in the SpikeDataLoading\_openEphys\_KiloSort\_A1.m script
2. Before running the 2nd section, open the variable spikeClusterData.uniqueCodes and double-check at least for the good clusters the channel number (from Phy) in the 2nd column; make sure the channel number in Phy is correct (compare to the waveform view), sometimes it is displayed wrong
3. Run the other sections till the end of the script
4. In the PlotPSTHandRaster\_openEphys\_KiloSort\_A1.m script in Matlab, run the first 2 sections
5. In the displayed figures, select (click the radio button) the Good/SUA clusters with visual responses (VisEv), optogenetic effect (Spont) or no response/effect (none)
6. Run the next two sections
7. Run section 5 (Mua analysis) and select in each figure the Mua clusters with good visual response (VisEv) or no/poor visual response (none)
8. Run the next two sections
9. Run the next section with average of SUA (meanTrace, MeanAllCondTrace.fig) and average of Mua (meanTraceMua, MeanAllCondTraceMua.fig)
10. Open waveformAnalysis\_openEphys\_KiloSort\_A1.m, run the first section and make sure the waveforms look like proper spikes and the amplitude distribution is uni-modal
11. Run the next 2 sections
12. For the orientation protocol, run the oriAnalysis.m script

Which script creates which structure?

load\_command\_KiloSort\_A1.m : **sessionInfo.mat**, **timeSeries.mat**

SpikeDataLoading\_openEphys\_KiloSort\_A1.m : **spikeClusterData.mat**

PlotPSTHandRaster\_openEphys\_KiloSort\_A1.m : **clusterTimeSeries.mat**

waveformAnalysis\_openEphys\_KiloSort\_A1.m: **cellMetrics.mat**

Check the existence of the respective structures after each running each script.