Elyashiv Zangen 2021

User manual – cell display and clustering app final MATLAB project

Contents

Introduction

The project is divided in to 2 parts of different levels of analysis. The first part is a cell display app and the second is a clustering app.

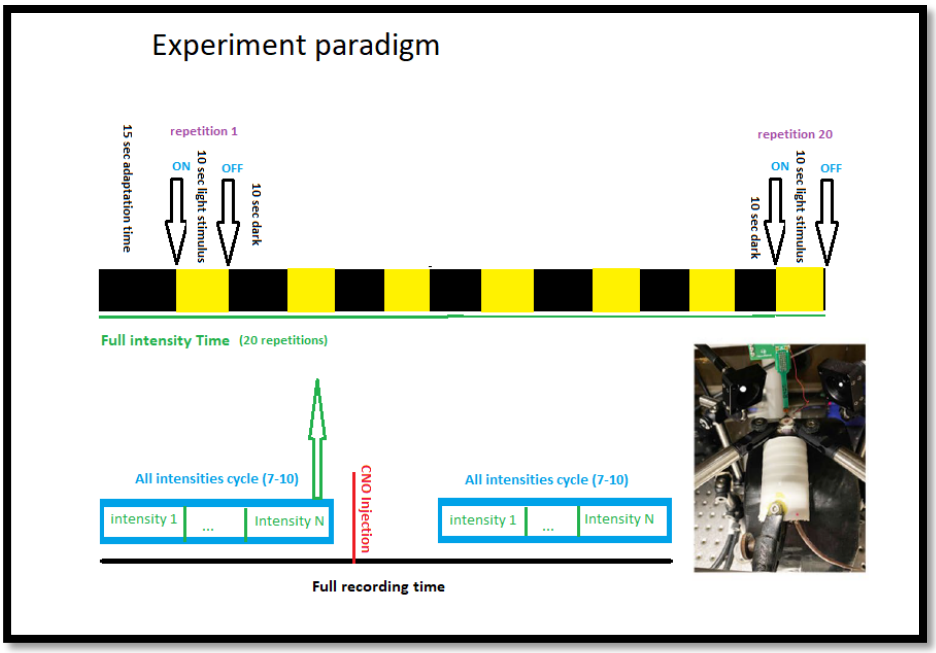
The goal of the cell display app is to analyze and display the results of a single extracellular electrophysiology experiment using a 32 channel multielectrode (but is useful for different kind of electrodes). The experiment goal is to record the responses to a light stimulus of deferent intensities before and after a chemogenetic silencing of a brain Region (parahabernular nucleus - PHB) the presumably regulates the light responses (the recording can be from the PHB or ather brain areas such as the prefrontal cortex or the nucleus accumbens). The app is not the first part of the analyzing process but it is based on a spike sorting of the kilosort program[1] and on a step of creating a raw PSTH form the output of the single neurons detected by kilosort. In addition, the app can use a file of cell positions from a prior analyzing that determines for each neuron in the recording in which brain Region is. Another option is using the raw spike times outputted form kilosort to show the changes of the firing rate Throughout the hole experiment.

The cell display app takes the raw PSTH and calculates and displays several parameters that help to understand the responses of each cell to light, the intensity dependence of the light responses (IR), and the effect of the chemogenetic silencing. Then the app saves all the parameters to a file containing all the cells of the experiment for later analysis of the hole population and for clustering.

The second part of the project is the clustering app that clusters the app in order to find different response type and identify clusters of distinct functional neurons. which takes a number of experiments after the analyzing of the cell display app and helps to tune the clustering parameters and clusters all the cells using density-based spatial clustering of applications with noise clustering (DBSCAN) or the gaussian mixture model clustering (GMM). The clustering is based on the PSTH waveform of the best response for each cell (after a PCA dimensionality reduction) as determent by the analyzing of the cell display app.

https://github.com/MouseLand/Kilosort [1]

The experiment the project analyzes:

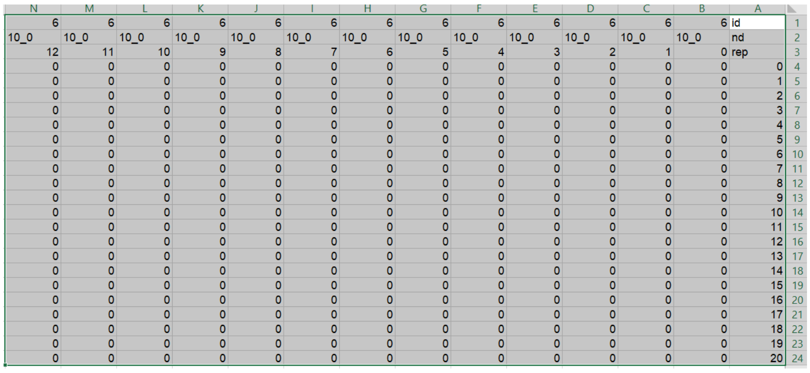


In the experiment we recorded acute extracellular recordings with a 32 multi-unit electrode in head-restrained mice while presenting a light stimulus, and injecting CNO to silence some brain regions (that were injected previously with DREEDDs). The light stimulus was presented at 7 different intensities from log form 9.4-15.4 log photons $cm^{-2}s^{-1}$ the intensity number is later referred as ND (where 1 is the highest (15.4) and 10 is the lowest (9.4)). Every intensity of the stimulus began with 15 seconds of darkness followed by 20 repetitions of a 10-second pulses of light and 10 seconds of darkness and then started the higher intensity. After the whole set of intensities and repetitions we injected CNO, started a short recording (20 minutes) of the silence part (with no stimulus) and then started a 7/10-intensity*20 repetitions again. The goal of the experiment is to analyze the responses to light with and without the injection silencing different brain regions.

A large part of the code for analyzing was already written in the lab so the cell GUI Is based on some data that is already partly analyzed: After the recording we filter the data will the high pass of 300HZ (plexon offline sorter) and the perform automatic spike sorting followed by manual revision with a MATLAB program and python tool (kilosort, Phy). Then we run a python code written in the lab that creates a CSV table that includes bind raw PSTH of all the neurons detected in spike sorting divided to 2 files – one before the CNO injection and one after it.

In the PSTH each cell detected in the spike sorting (id) process has 140 columns for each intensity (ND) and repetition (rep) (7*20). In each column the rows represent the firing rate of 19 seconds (with a 0.2 sec sampling rate[2]) - 3 pre-stimulus, 10 of stimulus and 6 post stimuli (the first and last bin were erased so it is 93 and not 95 bins):

[2] The sampling rate is in a very low resolution. It is indeed not enough to track the latency of the response to light. I used this sampling rate because all the data in my lab is already in that format and I wanted it to work with my data. For future data it is easy to use the app with raw PSTH's with higher resolution by changing the bin_size property in the app.



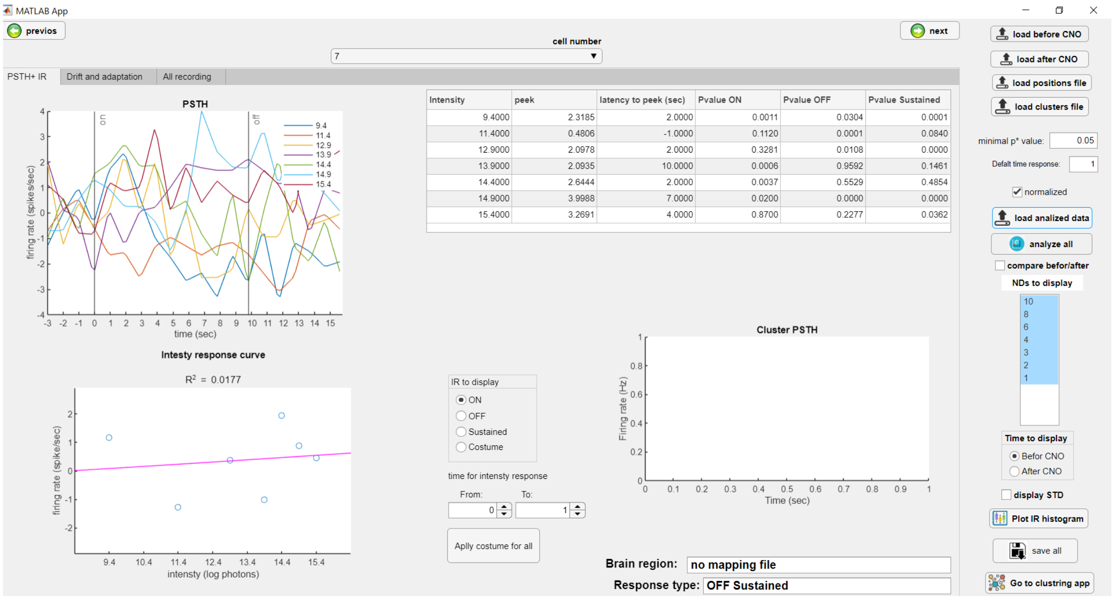
N	M	L	K	J	I	H	G	F	E	D	C	B	A	
6	6	6	6	6	6	6	6	6	6	6	6	6	id	1
10_0	10_0	10_0	10_0	10_0	10_0	10_0	10_0	10_0	10_0	10_0	10_0	10_0	nd	2
12	11	10	9	8	7	6	5	4	3	2	1	0	rep	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	5
0	0	0	0	0	0	0	0	0	0	0	0	0	2	6
0	0	0	0	0	0	0	0	0	0	0	0	0	3	7
0	0	0	0	0	0	0	0	0	0	0	0	0	4	8
0	0	0	0	0	0	0	0	0	0	0	0	0	5	9
0	0	0	0	0	0	0	0	0	0	0	0	0	6	10
0	0	0	0	0	0	0	0	0	0	0	0	0	7	11
0	0	0	0	0	0	0	0	0	0	0	0	0	8	12
0	0	0	0	0	0	0	0	0	0	0	0	0	9	13
0	0	0	0	0	0	0	0	0	0	0	0	0	10	14
0	0	0	0	0	0	0	0	0	0	0	0	0	11	15
0	0	0	0	0	0	0	0	0	0	0	0	0	12	16
0	0	0	0	0	0	0	0	0	0	0	0	0	13	17
0	0	0	0	0	0	0	0	0	0	0	0	0	14	18
0	0	0	0	0	0	0	0	0	0	0	0	0	15	19
0	0	0	0	0	0	0	0	0	0	0	0	0	16	20
0	0	0	0	0	0	0	0	0	0	0	0	0	17	21
0	0	0	0	0	0	0	0	0	0	0	0	0	18	22
0	0	0	0	0	0	0	0	0	0	0	0	0	19	23
0	0	0	0	0	0	0	0	0	0	0	0	0	20	24

Required packages for running:

1. npy-matlab: a package for loading the .npy kilosort structures to MATLAB
   https://github.com/kwikteam/npy-matlab
2. Curve Fitting Toolbox: a MATLAB toolbox for fitting
   https://www.mathworks.com/products/curvefitting.html
3. raacampbell/shadedErrorBar: function the creates shaded error bars of around the mean plot.
   https://github.com/raacampbell/shadedErrorBar
4. Statistics and Machine Learning Toolbox: MALAB toolbox for using the clustering algorithms:
   https://www.mathworks.com/products/statistics.html

Cell display GUI

Introduction:



cell number: 7

Intensity	peek	latency to peek (sec)	Pvalue ON	Pvalue OFF	Pvalue Sustained
9.4000	2.3185	2.0000	0.0011	0.0304	0.0001
11.4000	0.4806	-1.0000	0.1120	0.0001	0.0840
12.9000	2.0978	2.0000	0.3281	0.0108	0.0000
13.9000	2.0935	10.0000	0.0006	0.9592	0.1461
14.4000	2.6444	2.0000	0.0037	0.5529	0.4854
14.9000	3.9988	7.0000	0.0200	0.0000	0.0000
15.4000	3.2691	4.0000	0.8700	0.2277	0.0362

R^2 = 0.0177

IR to display: ● ON ○ OFF ○ Sustained ○ Costume

From: 0 To: 1

Brain region: no mapping file

Response type: OFF Sustained

minimal p* value: 0.05

Defalt time response: 1

☑ normalized

□ compare befor/after

Time to display: ● Befor CNO ○ After CNO

□ display STD

The GUIs purpose is to take the before + after raw PSTH csv file, analyze the parameters to be presented and display and save the results.

Parameters analyzed for each cell before and after CNO injection

exists in both before and after raw PSTH's

1. **Mean + std baseline** for each intensity calculated as the mean of the time before the ON stimulus (first 14 bins in the raw PSTH).
2. **IR curve + fitting sigmoid**: To evaluate whether the cell encodes the light intensity the app computes a dose-response curve for the different intensities (intensity response = IR curve). In addition, it finds the best fitting sigmoid[3] and computes the coefficient of determination (R^2) to evaluate the fit to the sigmoid curve – with is an estimation for the intensity encoding capacity. The IR+R^2 are computed for the ON (when the light turns on), OFF (when light turns off) and sustained (The end of the time the light is on) responses. In addition, it is possible to manually choose the time of the response (costumed)
3. **Parameters calculated for each intensity:**
    a. **Significance of the responses** (OFF, ON, SUSTAIND) calculated by the P value (Using Wilcoxon rank sum test) of each response.
    b. **Mean + STD Baseline of repetitions** – the baseline firing rate of each repetition for the relevant intensity.
    c. **Mean + STD firing rate** of the total stimulus time (10 seconds the light is on) for each repetition.
    d. **PSTH (mean+std)** for each intensity
    e. **Peek** of the response (the heist value of the PSTH)

[3] sigmoid is the function type traditionally used to evaluate the ability to encode light intensity. For that reason, I use it even though some functions can fit the IR curves better. The IR is calculated in the sigmoid fit function (In the apps functions) – which uses the MATLAB fit function to find the best parameters for the sigmoid function: "Rmax*10^(n*x)/(10^(n*x)+10^(n*logK)".

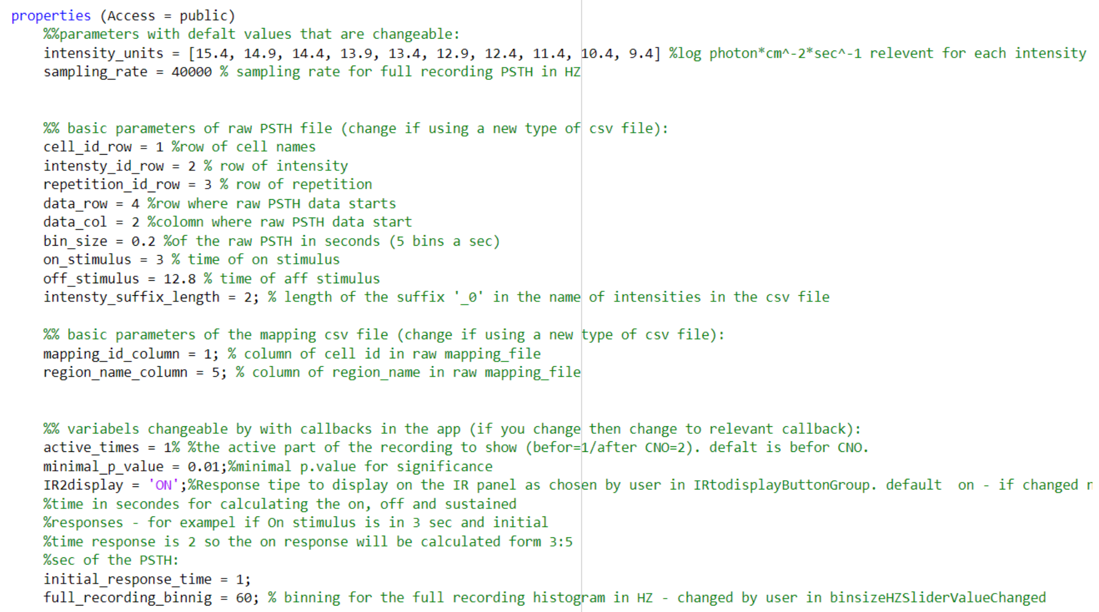
   f. **Latency to the peek** (time form the light onset)

4. **Maximum response** – the intensity that responded the best to light and what part of it responded best (on, off, sustained) calculated as the intensity and response with the lowest pValue.
5. **Type of the cell** - (ON/OFF/ON-OFF/not responding + sustained/not sustained) based on the Maximum responding intensity
6. **Cluster –** if user provides a GMM clusters file (produced by the clustering app) the app clusters each cell to the existing clusters based on the PSTH of the maximum responding intensity. That clusters file will also update according to all the cell in the experiment (Adding them to the existing clusters. For clustering again accurately use the clustering app). The cluster identity is calculated in the analyze all function – by expressing the PSTH waveform with the PCs originally used to perform the clustering in the clustering data.
7. **Cell position** – if user provides a cluster position csv file each cell is attached to the brain position form that file.
8. **Full experiment firing rate** – if user provides the spike_time.npy and spike_clusters.npy (output of Phy and kilosort) the app displays the firing rate of the cell (or average of all the cells in the experiment) over the course of the full experiment (including the injection time).

How to use – step by step

Step 0 – set parameters (optional)

Before we start the app there are some basic parameters that can be changed in the code (all of the variables are in the beginning of the properties section of the app code):



```
properties (Access = public)
    %%parameters with defalt values that are changeable:
    intensity_units = [15.4, 14.9, 14.4, 13.9, 13.4, 12.9, 12.4, 11.4, 10.4, 9.4] %log photon*cm^-2*sec^-1 relevent for each intensity
    sampling_rate = 40000 % sampling rate for full recording PSTH in HZ


    %% basic parameters of raw PSTH file (change if using a new type of csv file):
    cell_id_row = 1 %row of cell names
    intensty_id_row = 2 % row of intensity
    repetition_id_row = 3 % row of repetition
    data_row = 4 %row where raw PSTH data starts
    data_col = 2 %colomn where raw PSTH data start
    bin_size = 0.2 %of the raw PSTH in seconds (5 bins a sec)
    on_stimulus = 3 % time of on stimulus
    off_stimulus = 12.8 % time of aff stimulus
    intensty_suffix_length = 2; % length of the suffix '_0' in the name of intensities in the csv file

    %% basic parameters of the mapping csv file (change if using a new type of csv file):
    mapping_id_column = 1; % column of cell id in raw mapping_file
    region_name_column = 5; % column of region_name in raw mapping_file


    %% variabels changeable by with callbacks in the app (if you change then change to relevant callback):
    active_times = 1% %the active part of the recording to show (befor=1/after CNO=2). defalt is befor CNO.
    minimal_p_value = 0.01;%minimal p.value for significance
    IR2display = 'ON';%Response tipe to display on the IR panel as chosen by user in IRtodisplayButtonGroup. default  on - if changed r
    %time in secondes for calculating the on, off and sustained
    %responses - for exampel if On stimulus is in 3 sec and initial
    %time response is 2 so the on response will be calculated form 3:5
    %sec of the PSTH:
    initial_response_time = 1;
    full_recording_binnig = 60; % binning for the full recording histogram in HZ - changed by user in binsizeHZSliderValueChanged
```

   a. **initial_response_time** (default 1) – default time in seconds for calculating the on, off and sustained responses - for example if on stimulus is in 3 sec and initial time response is 1 so the on response will be calculated form 3:4.

b. **minimal_p_value** (default 0.01) - minimal p.value for significance

c. **intensity_units** – a list of all the intensities (in log photon*cm^{-2}*sec^{-1}) for the experiment type. Default value is for my 10 intensities data but can be changed according to needs. Should be ordered from high to low.

d. **IR2display** - Response type to display on the IR panel as chosen by user in IRtodisplayButtonGroup. default value is "on" - if changed need to change the default selection in IRtodisplayButtonGroup.

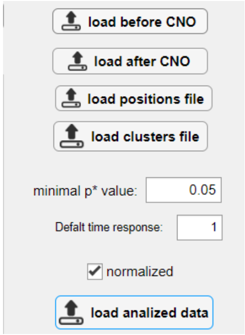
e. **basic parameters of histogram file.** (cell_id_row, intensty_id_row, repetition_id_row, data_row, data_col, bin_size, on_stimulus, off_stimulus, intensty_suffix_length). Change it only if using a new type of csv file.

f. **basic parameters of the mapping csv file** (mapping_id_column, region_name_column). Change it only if using a new type of csv file.

g. **sampling_rate** – sampling rate for the full recording (the kilosort files – not the raw PSTH. Default 40,000 Hz.

h. **Change parameters of the IR fit** – this is not in the app properties but in the sigmoid_fit function. The parameters used are experimentally good for my data but can be changed.

Step 1: load the files some are optional and some mandatory:



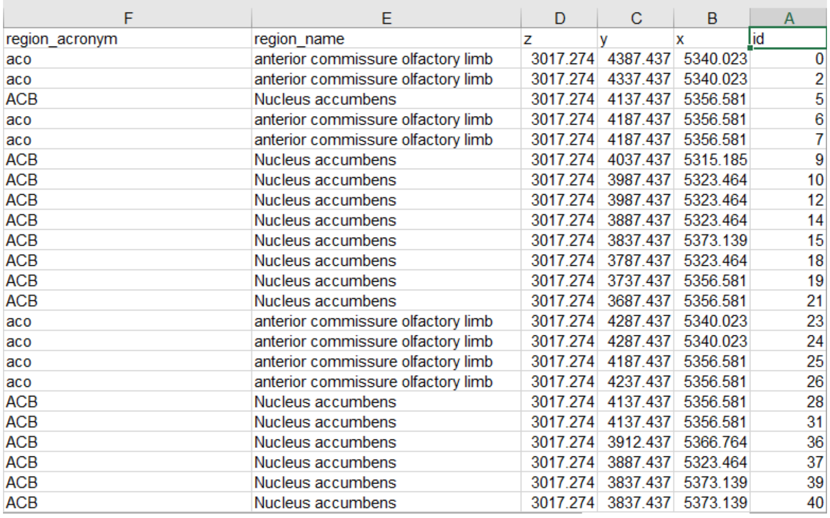
minimal p* value: 0.05

Defalt time response: 1

☑ normalized

1. **<span>(Mandatory)</span> Load before CNO file:** load the raw PSTH csv file as described above. The file name should be an identifiable name for the experiment because it will be used as the experiment id in the clusters output (and as default name while saving the analyzed data).
   The load button callback loads the csv file to a matrix. From the matrix it extracts the intensities and cell numbers relevant for the experiment and updates the GUI cell number drop down and NDs to display.
2. **(optional) load after CNO file**: load the raw PSTH csv file as described above. The callback updates the cell numbers to contain only cells that were detected (by kilosort) before and after the injection. If you want to analyze all the cells you need to load one file at a time (as a before CNO injection) and analyze and save them separately (not recommended because cells that where lost are probably not detected well by kilosort).
3. **(optional) load positions file**: a csv file with all the cell numbers with the brain Region containing each of the cells (csv that we create analyzing the track of the electrode with

sharp track MATLAB tool[4] and python scrips from are lab). This part is optional and if you load it the app will save the position of each cell and display it in the brain region field. The mapping csv file should look like this:



F	E	D	C	B	A
region_acronym	region_name	z	y	x	id
aco	anterior commissure olfactory limb	3017.274	4387.437	5340.023	0
aco	anterior commissure olfactory limb	3017.274	4337.437	5340.023	2
ACB	Nucleus accumbens	3017.274	4137.437	5356.581	5
aco	anterior commissure olfactory limb	3017.274	4187.437	5356.581	6
aco	anterior commissure olfactory limb	3017.274	4187.437	5356.581	7
ACB	Nucleus accumbens	3017.274	4037.437	5315.185	9
ACB	Nucleus accumbens	3017.274	3987.437	5323.464	10
ACB	Nucleus accumbens	3017.274	3987.437	5323.464	12
ACB	Nucleus accumbens	3017.274	3887.437	5323.464	14
ACB	Nucleus accumbens	3017.274	3837.437	5373.139	15
ACB	Nucleus accumbens	3017.274	3787.437	5323.464	18
ACB	Nucleus accumbens	3017.274	3737.437	5356.581	19
ACB	Nucleus accumbens	3017.274	3687.437	5356.581	21
aco	anterior commissure olfactory limb	3017.274	4287.437	5340.023	23
aco	anterior commissure olfactory limb	3017.274	4287.437	5340.023	24
aco	anterior commissure olfactory limb	3017.274	4187.437	5356.581	25
aco	anterior commissure olfactory limb	3017.274	4237.437	5356.581	26
ACB	Nucleus accumbens	3017.274	4137.437	5356.581	28
ACB	Nucleus accumbens	3017.274	4137.437	5356.581	31
ACB	Nucleus accumbens	3017.274	3912.437	5366.764	36
ACB	Nucleus accumbens	3017.274	3887.437	5323.464	37
ACB	Nucleus accumbens	3017.274	3837.437	5373.139	39
ACB	Nucleus accumbens	3017.274	3837.437	5373.139	40

4. **(optional) load clusters file:** A .mat file created by the clustering app. works only for cluster files of GMM clustering (in the DBSCAN you need to cluster everything again in the clustering app). Later in the analyzing function the app will cluster each cell to the best cluster (gaussian model) and save the cells to the clusters file.
5. **(optional) load analyzed data:** because the analyzing can be long there is an option to load an old analyzed data file (created by the cell display app) to just view the data. Loading the analyzed data file without the before injection should work. Note that after loading it you cannot use analyze all and the next steps are irrelevant.

Step 2: set the parameters

this step is **optional** only if you want to change the basic parameters.

6. **(optional) normalized check button:** default on. determines if the PSTH of each intensity will be normalized to the baseline firing rate (subtract it from all the PSTH)**.**
7. **(optional) adjust minimal p*value:** (default 0.01) minimal p.value for significance the on, off, sustained and costume responses
8. **(optional) change default time response:** default time in seconds for calculating the on, off and sustained responses - for example if on stimulus is in 3 sec and initial time response is 1 so the on response will be calculated form 3:4.
9. **(optional) NDs to display:** After loading the before CNO file all the relevant NDs will be selected (the NDs refer to the intensities of the light stimulus where 1 is the highest intensity and 10 is the lowest (15.4 – 9.4 log photons*cm^{-2}*sec^{-1})). It is possible to choose only some of them for analyzing (but not recommended – if you don't analyze all of them the app won't let you display those intensities you did not analyze – and they won't be part of the IR curve and fit). The NDs to display box can be used later on after analyzing the data.

[4] https://github.com/cortex-lab/allenCCF

Step 3: Analyze all

10. **(mandatory) run analyze all:** This step callback is a very long function that analyzes all the parameters noted above in the introduction (except of the full experiment firing rate which is a very heavy calculation that is not always needed and is analyzed separately), saves them to the app.all_data variable (which can later be saved), and displays them. This step is computationally heavy and can take some time. You will know it is done when all the data will be displayed.

Step 4: Look at the data and adjust parameters. We will go over all control buttons and the elements displayed:

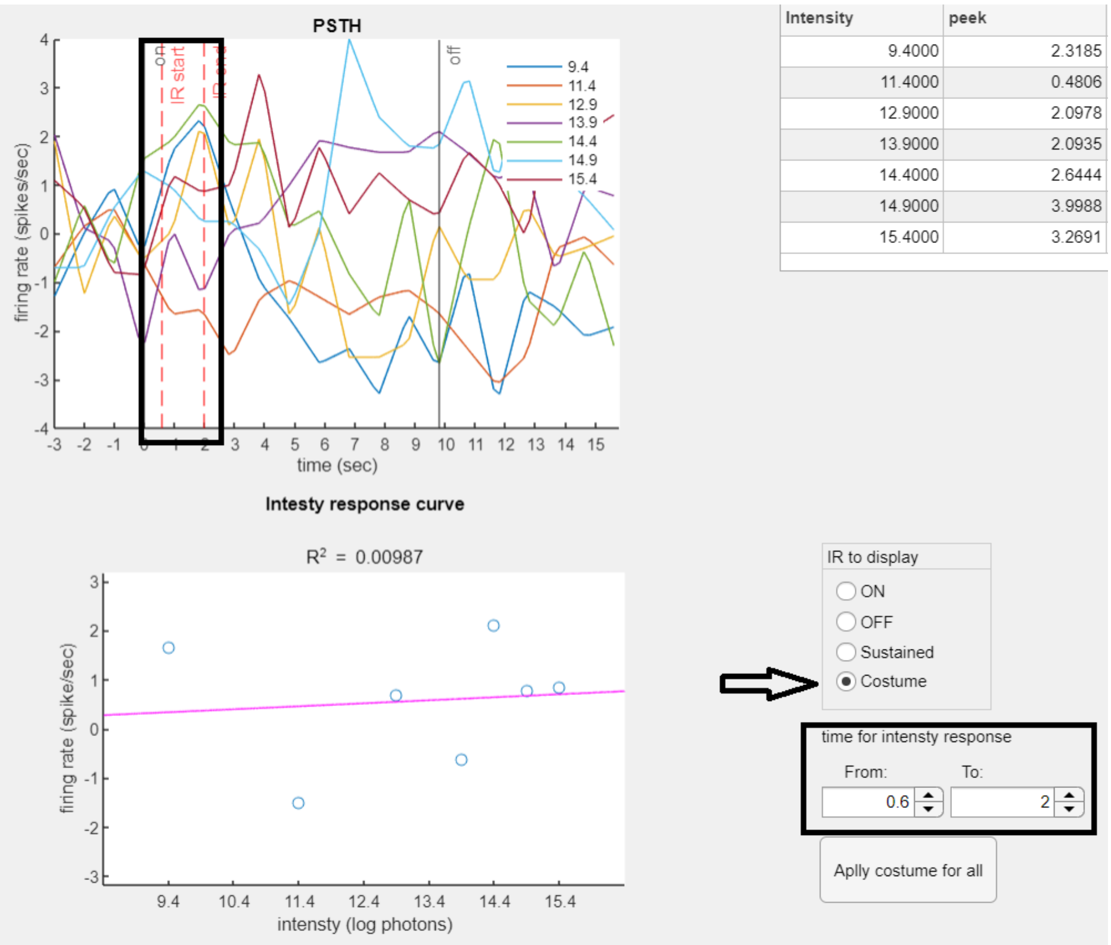
Choose parameters to display (the display will update automatically):

1. **NDs to display:** It is possible to control the intensities to be displayed. You can choose one ND or multiple NDs (by holding the shift button on the keyboard). If the compere before and after is checked multiple selection will be denied. (Changing the ND will not change the Intensity response curve or the intensity baseline).
2. **Time to display:** control if to display before or after CNO injection
3. **Change cell number:** use the dropdown menu or the next and previous buttons
4. **Display std:** When checked displays the standard deviation in all the plots (excluding for the repetition mean firing rate because the firing rate fluctuations during the light stimulus are very high).
5. **Compare before/after:** when this button is checked the PSTH, Repetition baseline, Repetition stimulus and intensity baseline plot the before and after of one intensity. by default, after checking the box it will be the highest intensity and then it is possible to change the intensity displayed but the multiselect option is off. This option allows the user to see easily the effect of the injection. The intensity response, cluster PSTH, response type field and table will still display only the time selected in Time to display.

6. **Plot IR histogram:** this button is used to assess the effect of the CNO ijection on the intensity encoding of the hole population represented by R^2. Pushing the button starts a function that plots on a new figure the mean of the R^2 and histogram for all cells in the experiment (it is presented on a new figure because I do not use this feature much and don't want to make the app heavier).
7. **IR to display[5]:** Control the part of PSTH used for the IR curve as explained in the introduction. The ON, OFF and Sustained are calculated in the analyze all function. The costume allows the user to choose the time for the IR and fit calculation by changing the time for intensity response (From: and To: fields). The time it those field correspond to the time from the light onset (0). When the costume field is selected 2 red lines appear on the PSTH plot that correspond to the values of "From" and "To" fields. Changing them will update the lines and calculate the new IR for the chosen PSTH part (between the red lines). This step can be very slow (the fit algorithm is computationally heavy) so it is recommended to use the Apply costume for all button to analyze all the cells with the costumed IR and make the moving between cells faster. In addition, moving the From

[5] Currently the legend of the plot is of (commented in the code) because understanding the plot is pretty intuitive and the legend cand hide some of the plot.

and To computes the IR each time so it is not recommended to change it rapidly because the app won't have time to calculate – so it is better to insert a number to the fields manually (and not use the spinner fast). Note that the new costume fit is calculated on the active intensities selected in "ND to display" so if less than 3 intensities will be selected the fit function won't work.



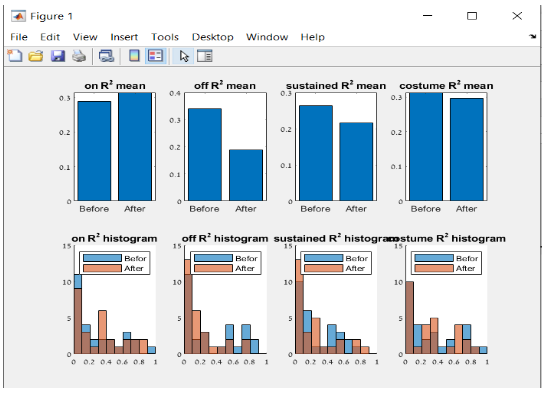
Intensity	peek
9.4000	2.3185
11.4000	0.4806
12.9000	2.0978
13.9000	2.0935
14.4000	2.6444
14.9000	3.9988
15.4000	3.2691

IR to display
○ ON
○ OFF
○ Sustained
● Costume

time for intensty response
From: 0.6 To: 2

Aplly costume for all

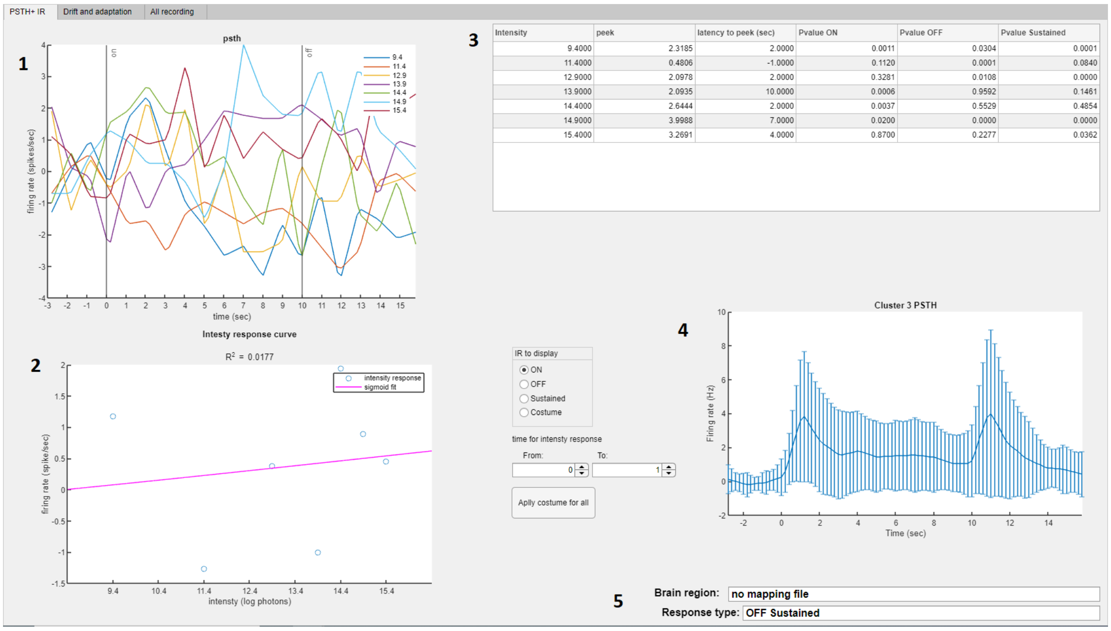
8. **Apply costume for all:** hitting this button saves to the app.all_data (in a new field) the IR + fitting sigmoid +R^2 for the PSTH part chosen. This can help iterating throe the cells firstly and use "plot IR histogram" on the costumed R^2.
9. **Plot IR histogram:** pressing the Plot IR histogram opens a new figure containing a histogram of the R^2 distributions and mean of all the cells in the experiment before and after injection for on, off sustained and custom:



Look at all the data:

all the data is plotted in the 2 tabs - PSTH + IR and drift and adaptation. (All recording is showing different data and will be explained later). All the data is plotted by the cell_display function (the IR is a different IR_plot function that is also used by cell_display).

TAB 1 - PSTH + IR:



Intensity	peek	latency to peek (sec)	Pvalue ON	Pvalue OFF	Pvalue Sustained
9.4000	2.3185	2.0000	0.0011	0.0304	0.0001
11.4000	0.4806	-1.0000	0.1120	0.0001	0.0840
12.9000	2.0978	2.0000	0.3281	0.0108	0.0000
13.9000	2.0935	10.0000	0.0006	0.9592	0.1461
14.4000	2.6444	2.0000	0.0037	0.5529	0.4854
14.9000	3.9988	7.0000	0.0200	0.0000	0.0000
15.4000	3.2691	4.0000	0.8700	0.2277	0.0362

IR to display
● ON
○ OFF
○ Sustained
○ Costume

time for intensty response
From: 0
To: 1

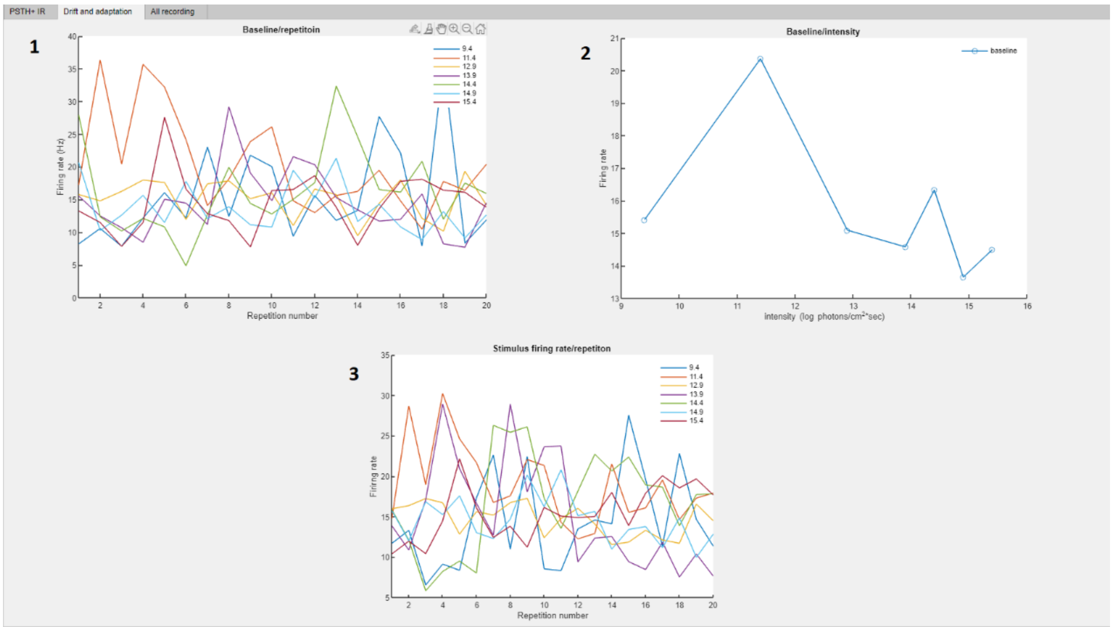
Aplly costume for all

Brain region: no mapping file
Response type: OFF Sustained

1. **PSTH plot –** show the mean PSTH for each intensity selected. The grey bars are the time of the stimulus on and off
2. **Intensity response curve –** presents all the intensities that where selected in the analyzing stage. Controlled by the IR to display buttons
3. **Table –** containing the statistical analyzing parameters for each intensity selected id ND to display: peek, latency to peek, Pvalue ON, Pvalue Off, Pvalue sustained – (p values were calculated Using Wilcoxon rank sum test)
4. **Cluster –** plot of the mean+-std (always shows the STD – does not depend on std display check box)[6]. It plots the mean of all cells in the cluster including the cells from the current experiment, cells from older experiments and cells clustered when first finding the clusters in the clustering app. (Note that each time analyzing adds all the cells of the experiment to the clusters file – so don't use a cluster file that contains the cells of your experiment and if you do don't save it and overwrite the same cluster because the mean won't be accurate and will contain a higher weight for the cells from the current cluster)**.** This will only be plotted if a clusters file was loaded.
5. **Brain region –** The full name of the brain region (if a position file is loaded)
   Response type – ON/OFF/ON-OFF/not responding + sustained/not sustained. Determined based on the significance of the pVlues for ON, OFF and sustained responses for the best responding intensity. Calculated in the analyze all callback.

[6]The clustering is based on the PSTH of the best responding intensity analyzed in analyze_all. The function takes only the relevant part of the response that was used for the original clustering (cuts most of the edges before and after the 10 sec stimulus). Uses the preprocessing used for the original (expressed by the same PCs and normalized to bounds) and then clusters using the same GMMs.

Tab 2 – drift and adaptation



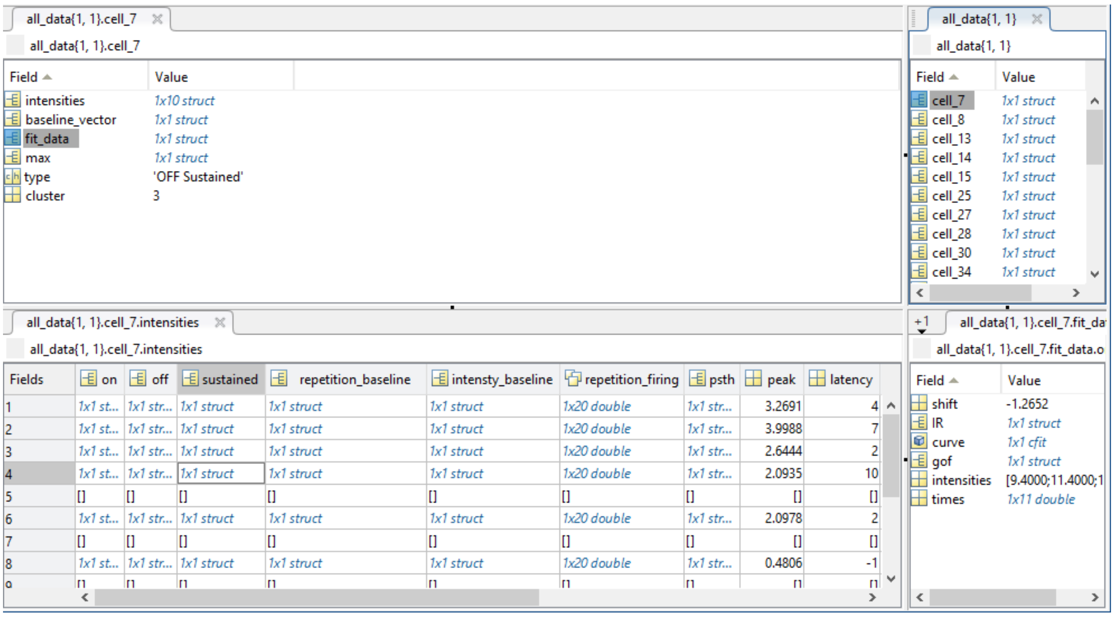
this tab shows the changes of the firing rate true the time of the recording (repetitions and intensities) and can show if there is a long-term adaptation or accumulation of the light. Another function of this Tab is to check if there are sudden drifts in the firing rate what can be a result of movement of the electrode that kilosort did not recognize well. Another goal of this tab is to compare before and after CNO injection and see the changes in baseline firing rate:

1. **Baseline/repetition**: The change of firing rate between repetitions. Calculated from the pre - normalized PSTH before the light turn on (3 first sec).
2. **Baseline/Intensity**: The change of the firing rate between intensities calculated as the mean of base line of all the repetitions baseline for each intensity.
3. **Stimulus firing rate/ repetition:** The mean firing rate of each repetition for the 10 seconds between the light on and light off.

Step 5: save the data

Clicking the save all button opens a dialog box to save 2 MATLAB structures containing all the data analyzed.

1. **App.all_data** – a structure containing all the data of the cells form the experiment. This file can be used for loading and displaying again in the cell display app, for the clustering app or for further analyzing. It contains 2 structs (one if you load only the before raw histogram) one for each time of the injection. Each struct contains fields corresponding to all the cell if the experiment. Each cell fields contains all the parameters calculated:



all_data{1, 1}.cell_7

Field	Value
intensities	*1x10 struct*
baseline_vector	*1x1 struct*
fit_data	*1x1 struct*
max	*1x1 struct*
type	'OFF Sustained'
cluster	3

all_data{1, 1}

Field	Value
cell_7	*1x1 struct*
cell_8	*1x1 struct*
cell_13	*1x1 struct*
cell_14	*1x1 struct*
cell_15	*1x1 struct*
cell_25	*1x1 struct*
cell_27	*1x1 struct*
cell_28	*1x1 struct*
cell_30	*1x1 struct*
cell_34	*1x1 struct*

all_data{1, 1}.cell_7.intensities

Fields	on	off	sustained	repetition_baseline	intensty_baseline	repetition_firing	psth	peak	latency
1	*1x1 st...*	*1x1 str...*	*1x1 struct*	*1x1 struct*	*1x1 struct*	*1x20 double*	*1x1 str...*	3.2691	4
2	*1x1 st...*	*1x1 str...*	*1x1 struct*	*1x1 struct*	*1x1 struct*	*1x20 double*	*1x1 str...*	3.9988	7
3	*1x1 st...*	*1x1 str...*	*1x1 struct*	*1x1 struct*	*1x1 struct*	*1x20 double*	*1x1 str...*	2.6444	2
4	*1x1 st...*	*1x1 str...*	*1x1 struct*	*1x1 struct*	*1x1 struct*	*1x20 double*	*1x1 str...*	2.0935	10
5	[]	[]	[]	[]	[]	[]	[]	[]	[]
6	*1x1 st...*	*1x1 str...*	*1x1 struct*	*1x1 struct*	*1x1 struct*	*1x20 double*	*1x1 str...*	2.0978	2
7	[]	[]	[]	[]	[]	[]	[]	[]	[]
8	*1x1 st...*	*1x1 str...*	*1x1 struct*	*1x1 struct*	*1x1 struct*	*1x20 double*	*1x1 str...*	0.4806	-1
9	[]	[]	[]	[]	[]	[]	[]	[]	[]

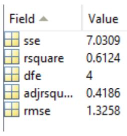
all_data{1, 1}.cell_7.fit_data.o

Field	Value
shift	-1.2652
IR	*1x1 struct*
curve	*1x1 cfit*
gof	*1x1 struct*
intensities	[9.4000;11.4000;1
times	*1x11 double*

a. **Intensities:** all table with all the parameters calculated for each intensity separately:
   i. **On/off/sustained** – 3 structures with the pvalue and the significance of the On/off/sustained response.
   ii. **Repetrition_baseline/repetiton_firing_rate** – stucts with mean+ std of the firing rate of stimulus/ baseline for each repetition.
   iii. **PSTH** -mean and std struct
   iv. Peak + latency (to peek)

b. **Basline vector –** struct with mean and STD of baseline firing rate/intensities

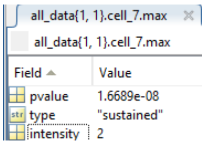
c. **Fit data –** a structure containing 3 identical structures for ON, OFF and Sustained (and one for costumed if apply custom for all was pressed)
   i. IR – intensity response curve (mean + std)
   ii. Shift – for calculation of the fit the IR was shifted by the minimum value so it won't be negative (with is a problem for the fitting algorithm)**.** This shift is needed for plotting the fitting curve in the wright position.
   iii. Curve – the fitting function
   iv. gof – a structure with statistics from the fitting algorithm including rsqure (R^2- coefficient of determination)



Field	Value
sse	7.0309
rsquare	0.6124
dfe	4
adjrsqu...	0.4186
rmse	1.3258

   i. intensities - the intensities (in log photon units) used for calculating the fit
   ii. times – the time of the response the IR is calculated from.

d. **Max –** structure with max responding intensity, response type and the pValue corresponding to that response (and it is the minimal pValue of the cell)



all_data{1, 1}.cell_7.max

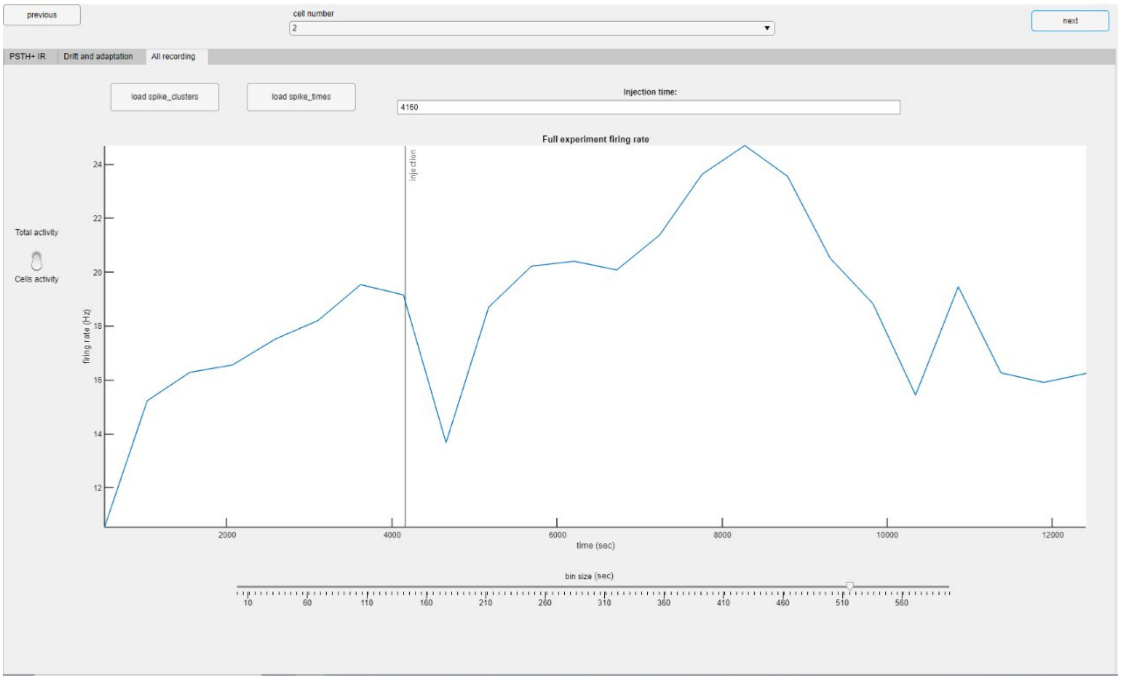
Field	Value
pvalue	1.6689e-08
type	"sustained"
intensity	2

   e. **Type –** type of cell (as plotted)
   f. **Cluster –** number of clusters the cell was clustered in. -only exists if clusters are loaded
   g. **Position** (not showed in the picture) – the brain Region of the cell. only exists if brain position is loaded

2. **App.clusters –** the updated clusters data with all the new cells (will be presented in the clustering app section)**.**

Step 6 – (optional) analyze all recording

this part is recommended to be last because it loads very heavy files that make the hole program very slow. This tab purpose is to plot a PSTH of the hole experiment and see the affect of the injection (the stimulus). To use it you need to load the files and use the controls.



cell number: 2

Injection time: 4160

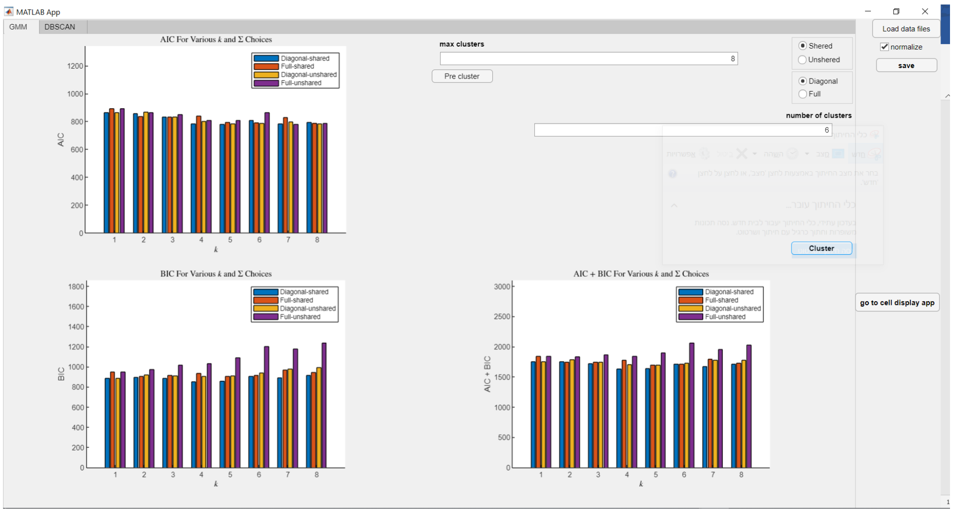
1. **Load the files –** load spike times and spike clusters – 2 outputs of kilosort. The first is long vector with the times of all the spikes of the hole recording and the second is a same size long vector with the cell numbers corresponding to the spike time spikes. After loading spike times, the full PSTH will be displayed.
2. Set parameters – after setting each parameter the display will update (slowly):
   a. **Injection time –** insert the time of the CNO injection in seconds from the beginning of the recording.
   b. **Bin size –** a slider determining the bis size in seconds.
   c. **Total activity/cell activity switch – the** cell activity option shows the PSTH of the currently active cell (and can be changed with the drop down/ next-previous buttons). The total activity is the firing rate of all the cells in the recording and gives a very macro-scale picture. Note that the total activity

includes all spikes that kilosort detected – including spikes from cells that are probably noise or MUA (multiunit activity – a cluster of far cells and not a single neuron) – these cells are not part of the raw histogram that uses only single unit cells from kilosort and after manually checking in Phy.

Clustering GUI

Introduction

Clustering GUI takes as input multiple experiment files (output of cell display app) performs a dimensionality reduction on the PSTH waveform of all the cells. Then it helps the user to control and tune the clustering parameters before the clustering and uses GMM[7] or DBSCAN[8] to cluster all the cells, creates a clusters variable containing all of the clusters and cell corresponding to them and plots the clusters mean+-std PSTH and identify different types of responses to light.



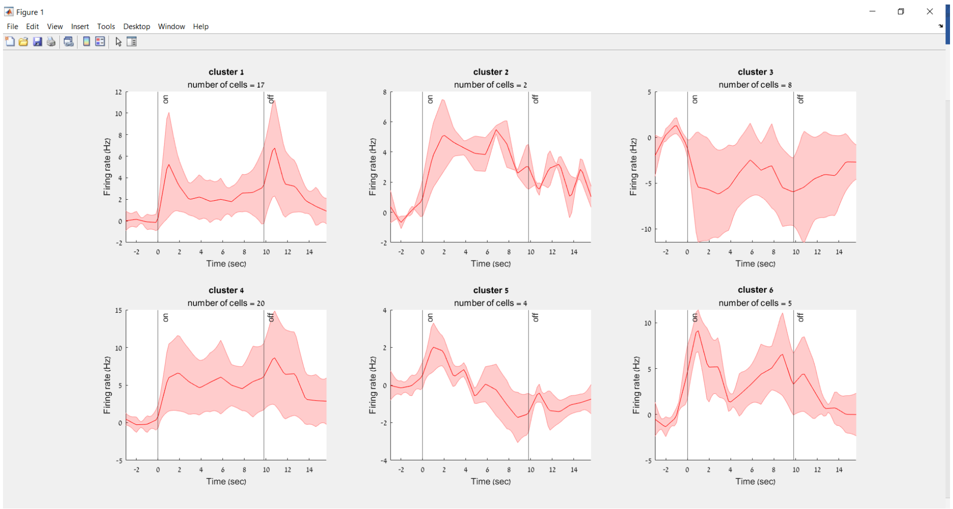
max clusters: 8

● Shered ○ Unshered

● Diagonal ○ Full

☑ normalize

number of clusters: 6



[7] https://www.mathworks.com/help/stats/clustering-using-gaussian-mixture-models.html

[8] https://www.mathworks.com/help/stats/dbscan-clustering.html#mw_0ad71796-4ae1-4620-a0e5-9ea5dcded2c6

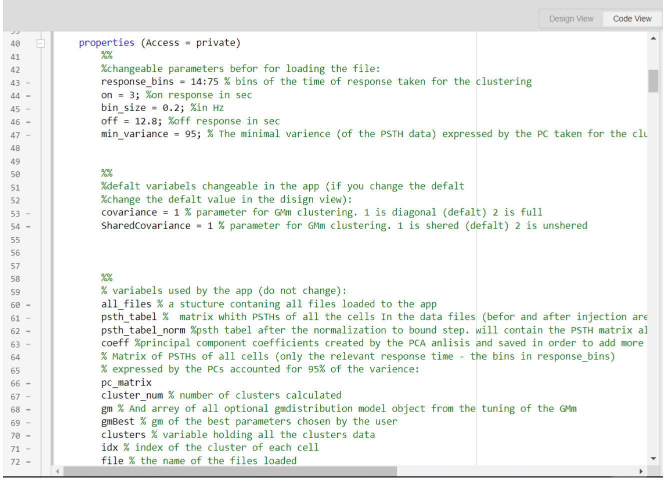
How to use – step by step

In this section we will go over how to use each feature of the app and explain how the use it and briefly how the analysis is done. Using the app has 3 basic steps – **load the data, tune the parameters and cluster, save the data**. Additionally, before using you can adjust the code parameters to fit your data.

STEP 0 (optional) – adjust the code parameters to fit your data

in the head of the code view there are parameters that can be adjusted and a short explanation to what they do. It includes:

1. Parameters that are related to the times of stimulus PSTH used (on, off, response bins, and bin size).
2. Min variance - A parameter controlling the minimal variance (of the PSTH data) expressed by the PC taken for the clustering – (which controls the number of PCs taken for the clustering and the tradeoff between accuracy of clustering and proceeding time).
3. Parameters for the GMM clustering that can be changed in the app (covariance and shared Covariance). Change only if you want to change the default value and change it in the design view as well.



```
properties (Access = private)
    %%
    %changeable parameters befor for loading the file:
    response_bins = 14:75 % bins of the time of response taken for the clustering
    on = 3; %on response in sec
    bin_size = 0.2; %in Hz
    off = 12.8; %off response in sec
    min_variance = 95; % The minimal varience (of the PSTH data) expressed by the PC taken for the clu


    %%
    %defalt variabels changeable in the app (if you change the defalt
    %change the defalt value in the disign view):
    covariance = 1 % parameter for GMm clustering. 1 is diagonal (defalt) 2 is full
    SharedCovariance = 1 % parameter for GMm clustering. 1 is shered (defalt) 2 is unshered



    %%
    % variabels used by the app (do not change):
    all_files % a stucture contaning all files loaded to the app
    psth_tabel %  matrix whith PSTHs of all the cells In the data files (befor and after injection are
    psth_tabel_norm %psth tabel after the normalization to bound step. will contain the PSTH matrix al
    coeff %principal component coefficients created by the PCA anlisis and saved in order to add more
    % Matrix of PSTHs of all cells (only the relevant response time - the bins in response_bins)
    % expressed by the PCs accounted for 95% of the varience:
    pc_matrix
    cluster_num % number of clusters calculated
    gm % And arrey of all optional gmdistribution model object from the tuning of the GMm
    gmBest % gm of the best parameters chosen by the user
    clusters % variable holding all the clusters data
    idx % index of the cluster of each cell
    file % the name of the files loaded
```

Step 1 – load the data

1. **(Optional) Set normalize ON/OFF**: (this should be done before loading the data) normalizing to bounds. If this box is checked after loading all PSTHs will be divided by the maximum absolute value which makes the bounds of the PSTH between 1 and -1. This option gives more weight to the PSTH waveform and less weight to the firing rate change magnitude.
2. **(Mandatory) Load all the files for clustering**: after hitting the load button a popup window will appear and you will need to choose the output structures of the cell display app (set them in the same folder before). The minimum files for clustering are 2 (1 file will not work). Make sure that the names of the files make sense – they will be used to tag the origin of the cell in the clusters output. After loading some preprocessing is going on:
    a. The PSTH of all the cells is extracted for the best intensity of each cell (in the LoaddatafilesButtonPushed callback). The best intensity was determined in

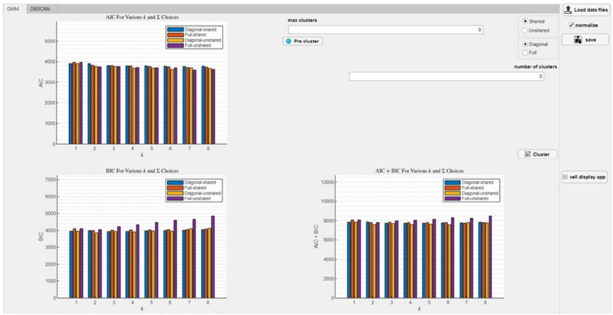
the cell display app as the cell with the lowest pvalue in one of the responses (ON, OFF, sustained) and almost always corresponds to ND = 1 or ND = 2 (the highest intensities).

b. A dimensionality reduction is performed by PCA (in Create_pc_matrix function) on the relevant bins of the PSTH as determined by response_bins. The PCs accountable for 95% of the variance are used to represent all the PSTHs for the clusters.

Step 2 – tune the clustering parameters

in this stage you need to choose the method for clustering – for each method the tuning is different and they need to be used separately and tune the parameters separately (and are in different tabs). You can use both methods, save both outputs and decide which method is better for your data:

1. **GMM - Gaussian Mixture Models** (in the GMM tab): This part option in using the GMM clustering (the hard clustering option) that fits a few GMM to the data and clusters the cells accordingly (by the probability of each cell PSTH to be a part of the model's distribution). the number of models is unknown and needs to be determined by the user. This method will not be explained in detail in this guide, which will only contain an explanation of how to adjust some basic parameters. The parameters that can be changed in this app are the number of clusters, and the covariance type (diagonal/full and shared/unshared). To find the best option:
   a. **Choose the maximal number of clusters in the max clusters field** (which should not be higher then 10 for a decent running time).
   b. **Hit the Pre cluster button**. The app will find the GMM fit for all 4-covariance option for all of number of clusters possible (1 – the max clusters chosen by the user), calculate the BIC (Bayesian Information Criterion) and AIC (Akaike Information Criterion) that asses the fitting of the model to the data and penalize for high number of clusters to avoid overfit. Then the app plots all the BIC, AIC and AIC+BIC for all the options to help the user choose the best option for clustering. (open the app in Fullscreen if the legend is hiding the plots)



max clusters 8

● Shered
○ Unshered

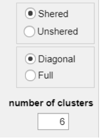
● Diagonal
○ Full

number of clusters 6

☑ normalize

   c. **Use the lowest BIC and AIC scored models to choose the best model for clustering** – by choosing the number the clusters and shared/unshared and full/diagonal. Take to note that the best model may not be clear and in order to find a good model you may need to check more than one option. Running the preculturing on the same data may resalt a slightly different AIC and BIC score each time. In addition, the BIC and AIC can point to deferent models.

BIC penalizes for complexity more severely than the AIC. Therefore, the AIC tends to choose more complex models that might overfit, and the BIC tends to choose simpler models that might underfit.

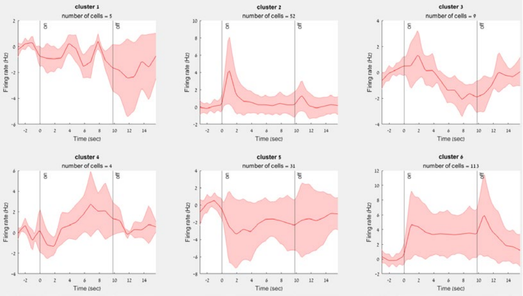


● Shered
○ Unshered

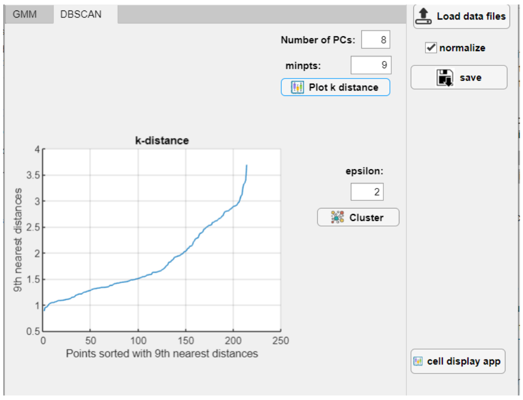
● Diagonal
○ Full

number of clusters: 6

d. **Press the cluster button** – the app will use the model chosen and cluster all the cells (from the PC matrix). Then the app will calculate (in plot_clusters function) the mean and STD of the PSTHs of all the cells for each cluster and plot them in e new figure (the shaded aria is + - the STD). After plotting you can change the parameters and cluster again to check more options.



2. **DBSCAN Density-based spatial clustering of applications with noise** (DBSCAN tab). A method cluster the cells basted on the distance (in the multidimensional PCs axes) but unlike k-mean it does not require prior knowledge of the number of clusters, and clusters are not necessarily spheroidal, and it can also cluster observations as noise (cell that are not part of any cluster)[9]. This method has 2 parameters to change – *minpts (*minimal neighbors for a cell to be a core of the cluster) and *epsilon (*the distance of 2 clusters to be neighbors) [10]. To use the DBSCAN follow:



Number of PCs: 8

minpts: 9

epsilon: 2

☑ normalize

a. **Choose minpts and plot k distance –** a good estimation for a good minpts value is the number of dimensions + 1 and that will be the default option chosen by the app (that finds the value while creating the PC matrix) but for

[9] https://www.mathworks.com/help/stats/dbscan-clustering.html#mw_0ad71796-4ae1-4620-a0e5-9ea5dcded2c6

[10] The DBSCAN algorithm identifies three kinds of points:
Core point — A point in a cluster that has at least minpts neighbors in its epsilon neighborhood
Border point — A point in a cluster that has fewer than minpts neighbors in its epsilon neighborhood
Noise point — An outlier that does not belong to any cluster

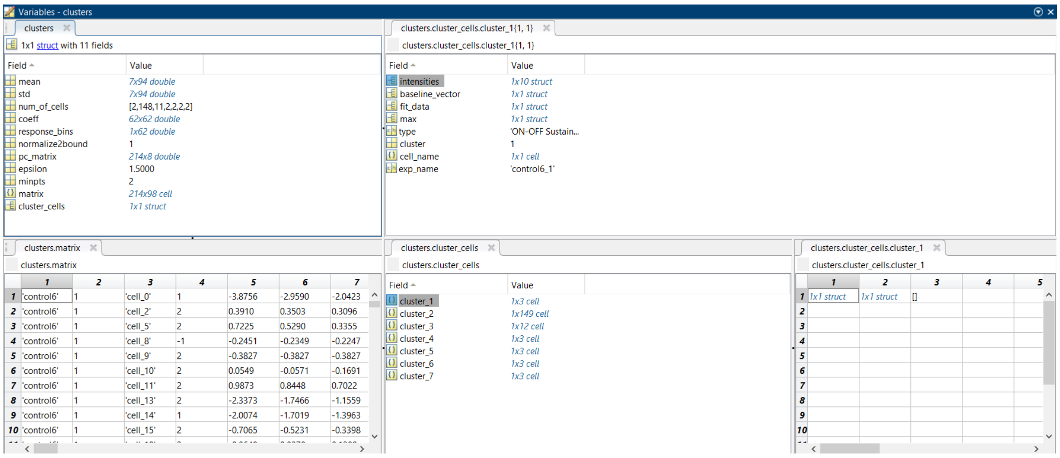
my data some times it is better to change it and check lower minpts options different options in order to get more clusters.

b. **Choose epsilon and cluster –** use the k-distance to estimate epsilon. The graph contains a knee. The distance that corresponds to the knee is generally a good choice for epsilon, because it is the region where points start tailing off into outlier (noise) territory. (For the example above an epsilon between 2-3 should be a good estimation).

c. **View the clusters and adjust parameters if needed–** for my data sets usually the BDSCAN gives a lower number of clusters (1-4) and if you what more clusters (by lowering minputs and epsilon) it can increase the number of cells that are outliers (clustered as noise).

Step 3 – save the data

saving the data will open a dialog box to specify a name for the clusters file. In addition, it will save the clusters for each cell in the original file loaded and save them in the same location (They will replace the old files). The culsters.mat will contain the data for all the cell clustered and can later be used for analyzing the clusters and cells. The file will be a bit different for the DBSCAN and the GMM and only the GMM can be used to later add cells to the clusters in the cell display app. The file has the following parameters:


Variables - clusters
clusters
clusters.cluster_cells.cluster_1{1, 1}
clusters.matrix
clusters.cluster_cells
clusters.cluster_cells.cluster_1

clusters

1x1 struct with 11 fields

Field	Value
mean	7x94 double
std	7x94 double
num_of_cells	[2,148,11,2,2,2,2]
coeff	62x62 double
response_bins	1x62 double
normalize2bound	1
pc_matrix	214x8 double
epsilon	1.5000
minpts	2
matrix	214x98 cell
cluster_cells	1x1 struct

clusters.cluster_cells.cluster_1{1, 1}

Field	Value
intensities	1x10 struct
baseline_vector	1x1 struct
fit_data	1x1 struct
max	1x1 struct
type	'ON-OFF Sustain...
cluster	1
cell_name	1x1 cell
exp_name	'control6_1'

clusters.matrix

	1	2	3	4	5	6	7
1	'control6'	1	'cell_0'	1	-3.8756	-2.9590	-2.0423
2	'control6'	1	'cell_2'	2	0.3910	0.3503	0.3096
3	'control6'	1	'cell_5'	2	0.7225	0.5290	0.3355
4	'control6'	1	'cell_8'	-1	-0.2451	-0.2349	-0.2247
5	'control6'	1	'cell_9'	2	-0.3827	-0.3827	-0.3827
6	'control6'	1	'cell_10'	2	0.0549	-0.0571	-0.1691
7	'control6'	1	'cell_11'	2	0.9873	0.8448	0.7022
8	'control6'	1	'cell_13'	2	-2.3373	-1.7466	-1.1559
9	'control6'	1	'cell_14'	1	-2.0074	-1.7019	-1.3963
10	'control6'	1	'cell_15'	2	-0.7065	-0.5231	-0.3398

clusters.cluster_cells

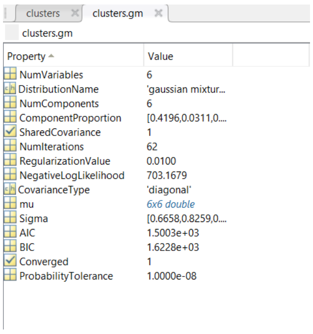
Field	Value
cluster_1	1x3 cell
cluster_2	1x149 cell
cluster_3	1x12 cell
cluster_4	1x3 cell
cluster_5	1x3 cell
cluster_6	1x3 cell
cluster_7	1x3 cell

clusters.cluster_cells.cluster_1

	1	2	3	4	5
1	1x1 struct	1x1 struct	[]		
2					
3					
4					
5					
6					
7					
8					
9					
10					

1. **Mean + std** - PSTH of all cells for each cluster.
2. **Numer_of_cells** - Number of cells in each cluster
3. **Coeff** – principal component coefficients created by the PCA analysis
4. **Response_bins + normalized2boound** – the bins of the PSTH used for the clustering and if it was normalized.
5. **PC_matrix** – matrix of all the PSTHs expressed as PCs from the PCA analysis.W
6. **Matrix** – a matrix of the PSTHs of all cells. The first 4 colomns are IDs for the experiment name ('control 6' in example – taken form the file name), time of the experiment (1=before injection 2 =after injection), cell name, and id of the cluster (-1 clusters are cells DBSCAN classified as noise).
7. **Cluster_Cells –** a structure containing all of the cells in each cluster. each cell contains all the data from app_dispaly including the cell name and experiment name.
8. **Epsilon+minpts** – only in DBSCAN the parameters used.

9. **GM –** only for GMM. A structure containing all the gm data of the best GMM used (output of the fitgmdist function).



Property	Value
NumVariables	6
DistributionName	'gaussian mixtur...
NumComponents	6
ComponentProportion	[0.4196,0.0311,0....
SharedCovariance	1
NumIterations	62
RegularizationValue	0.0100
NegativeLogLikelihood	703.1679
CovarianceType	'diagonal'
mu	*6x6 double*
Sigma	[0.6658,0.8259,0....
AIC	1.5003e+03
BIC	1.6228e+03
Converged	1
ProbabilityTolerance	1.0000e-08