Supplementary Material

NeuroActivityToolkit - toolbox for quantitative analysis of miniature fluorescent microscopy data

Mitenev A.V.1#*, Gerasimov E.I.1#*, Pchitskaya E.I.1, Chukanov V.S.1, Bezprozvanny I.B.1,2*

#equal contribution

*** Correspondence:**
Corresponding Authors: aleksandr.mitenev@mail.ru; evgeniigerasimov1997@gmail.com; ilya.bezprozvanny@utsouthwestern.edu

1 Supplementary Data

1. Installation

System requirements:

- Windows 8 or newer
- Minimum 1GB RAM
- Minimum 6 GB disk space

1.1. Installation steps:

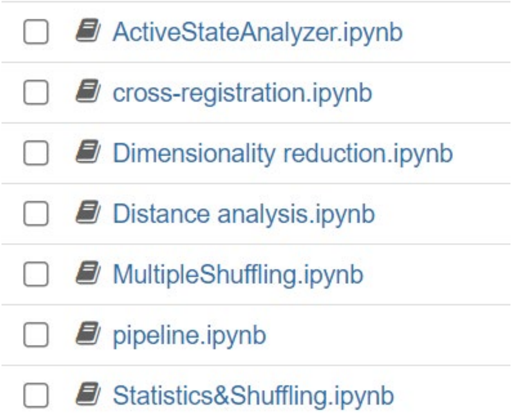
Download and install the minian package from: https://github.com/denisecailab/minian;
Download and unzip the code folder from: https://github.com/spbstu-applied-math/NeuroActivityToolkit or https://appliedmath.gitlab.yandexcloud.net/lmn/miniscope.
Install Anaconda and open Anaconda Promt;
Execute:

```
cd PATH_TO_CODE
conda create --name neuron-analysis -c conda-forge --file
requirements.txt –y
```

1. To run the software open Anaconda Promt, execute:

```
cd PATH_TO_CODE
conda activate neuron-analysis
jupyter notebook
```

After this steps, the joined Jupiter Notebook, containing separated notebooks, will be open:



- □ ActiveStateAnalyzer.ipynb
- □ cross-registration.ipynb
- □ Dimensionality reduction.ipynb
- □ Distance analysis.ipynb
- □ MultipleShuffling.ipynb
- □ pipeline.ipynb
- □ Statistics&Shuffling.ipynb

Click on required notebook to start work. Notebook descriptions are provided below.

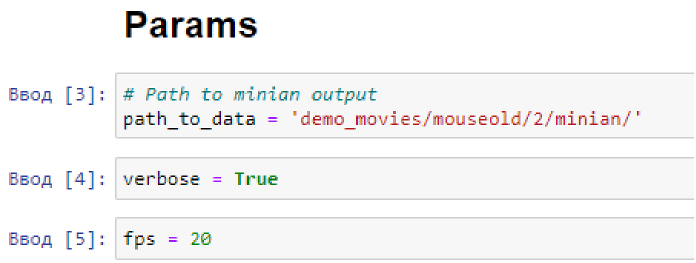
2. Example dataset

Example dataset contains of 3 recording for the same FVB mouse aged 9 months. The data is raw recordings for different days in ".avi" format that are ready to be processed via "Minian"

3. "ActiveStateAnalyzer" module

This Notebook is used to process the results of the "Minian" software package or others with the same output. The main role of this module is to compute active states of the individual neurons by different methods and to obtain metrics that are connected to activation parameters of the cells.

Parameters:



Params

```
Ввод [3]: # Path to minian output
          path_to_data = 'demo_movies/mouseold/2/minian/'

Ввод [4]: verbose = True

Ввод [5]: fps = 20
```

In this step parameters of miniscope data are set:

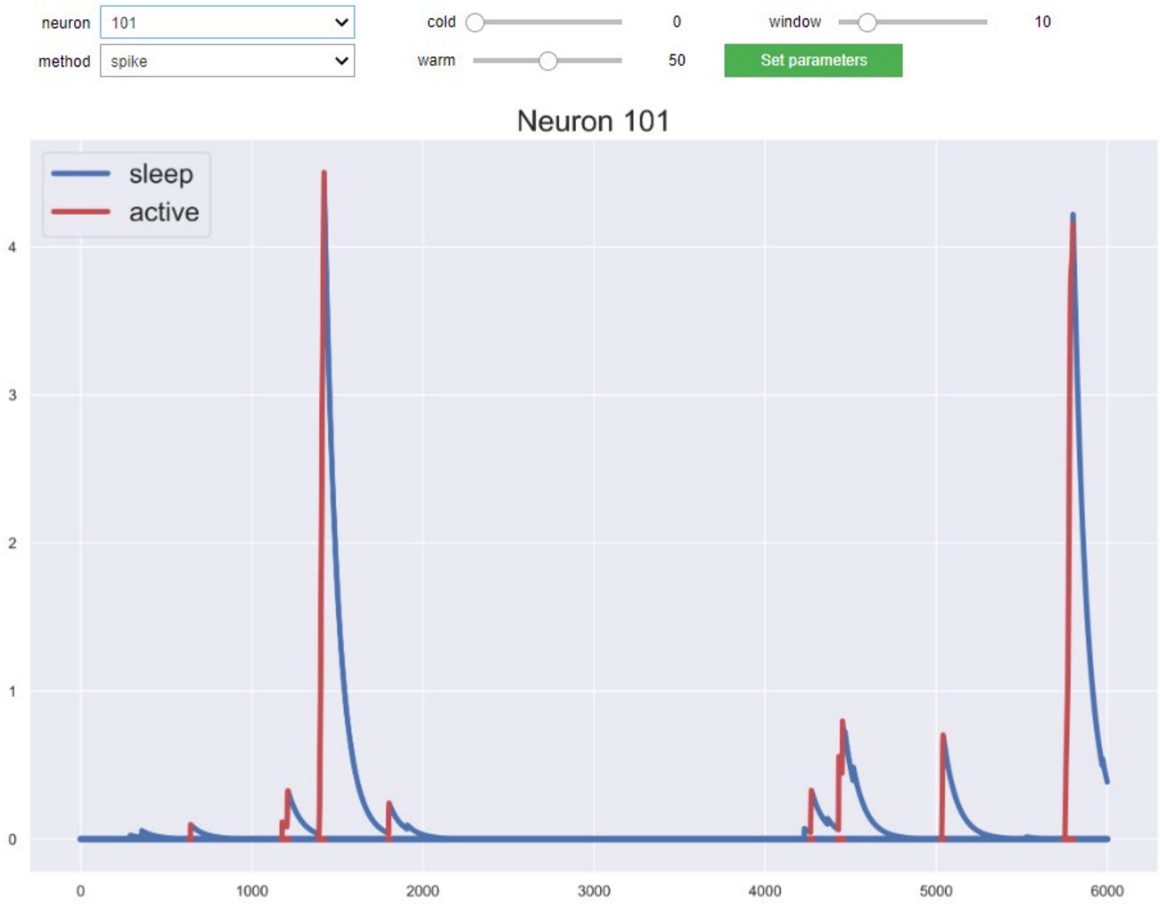
- path_to_data – way to processed data via "Minian";
- verbose – define visualization of progressbars;
- fps – frames per seconds for recording (important to set right value as all metrics are based on this value).

3.1. Determination of the active state of neuron

The possibility of setting parameters for detecting the active state of a neuron is presented:

- neuron – drop-down list with neuron selection;
- method – a way to detect active phase:
  - spike – only a stage of rapid intensity growth,

  - full – a stage of growth and decline of the intensity which is higher than calculated threshold value.
- cold – minimum duration of the active phase;
- warm – minimum duration of the passive phase;
- window – width of the moving average smoothing window;
- Set parameters – a button for setting the selected parameters and applying them to all neurons (highly important to “set parameters”, without it program code won’t work).



neuron 101

method spike

cold 0

warm 50

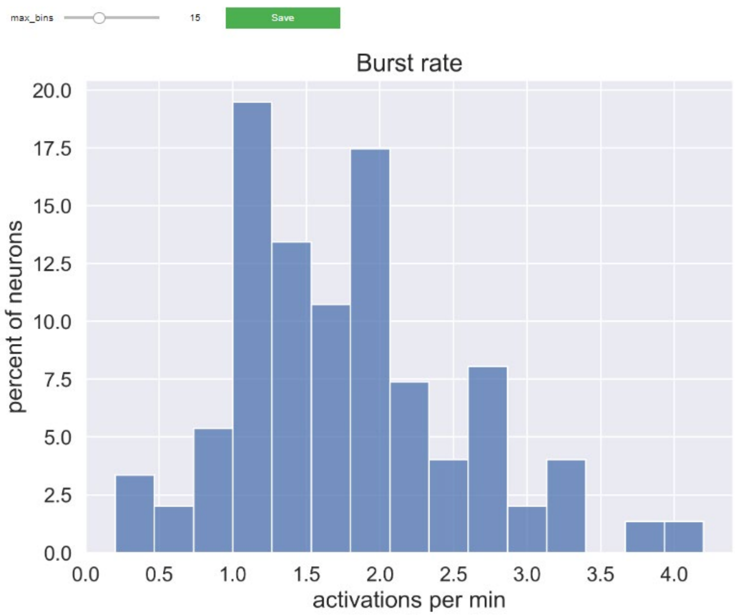
window 10

3.2. Burst rate

Burst rate – the number of "cell activations" for a given period of time.

The distribution of burst rate values is visualized.

- max_bins – maximum number of columns on the chart;
- Save – saving the distribution in excel.



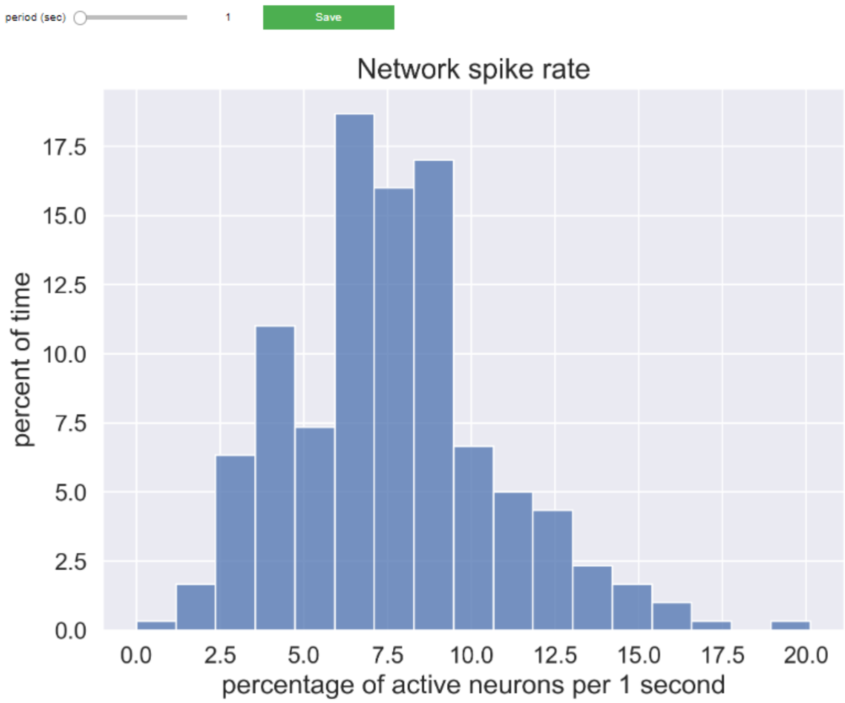
max_bins 15

3.3. Network spike rate and Network spike peak

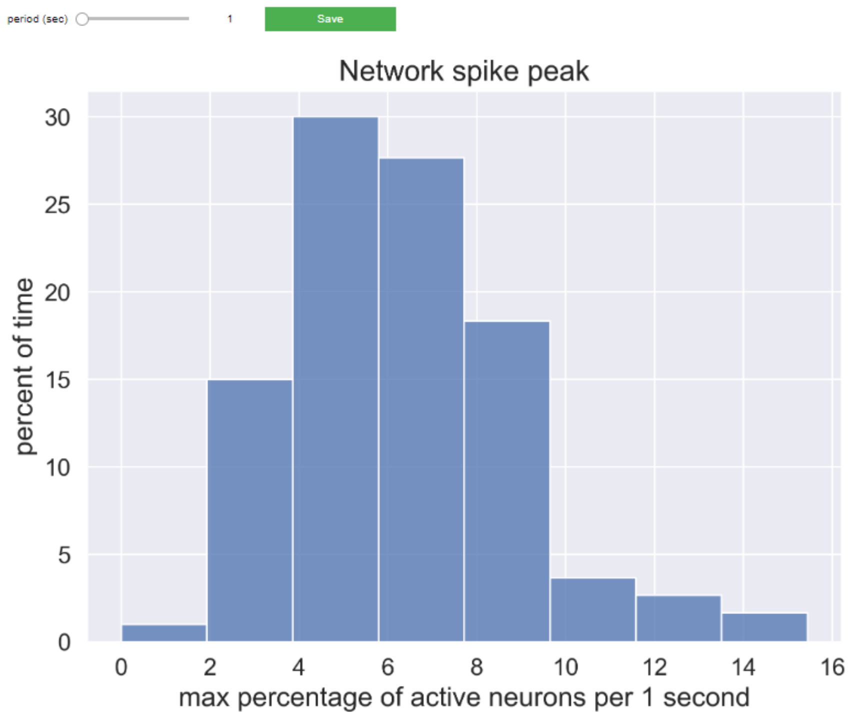
Network spike rate – the proportion of active neurons over a certain period of time;

Network spike peak – the maximum proportion of simultaneously active cells for a certain period of time

- period – duration of the interval for which metric will compute results (1sec-60sec);
- Save – saving the distribution in excel.



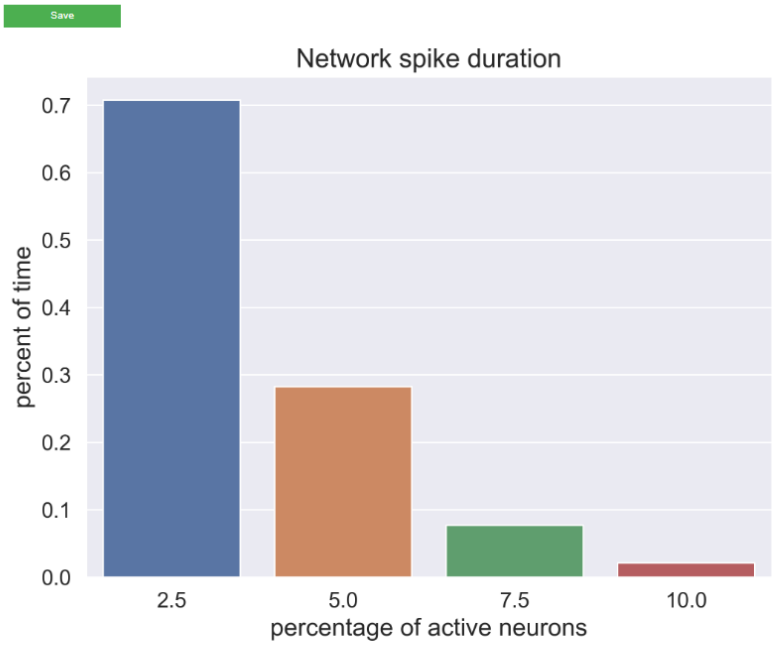
period (sec) 1



3.4. Network spike duration

Network spike duration – the length of time when the number of active cells is higher than the preset threshold number of neurons, computed results can be found in ".xlsx" from Network spike rate metric;

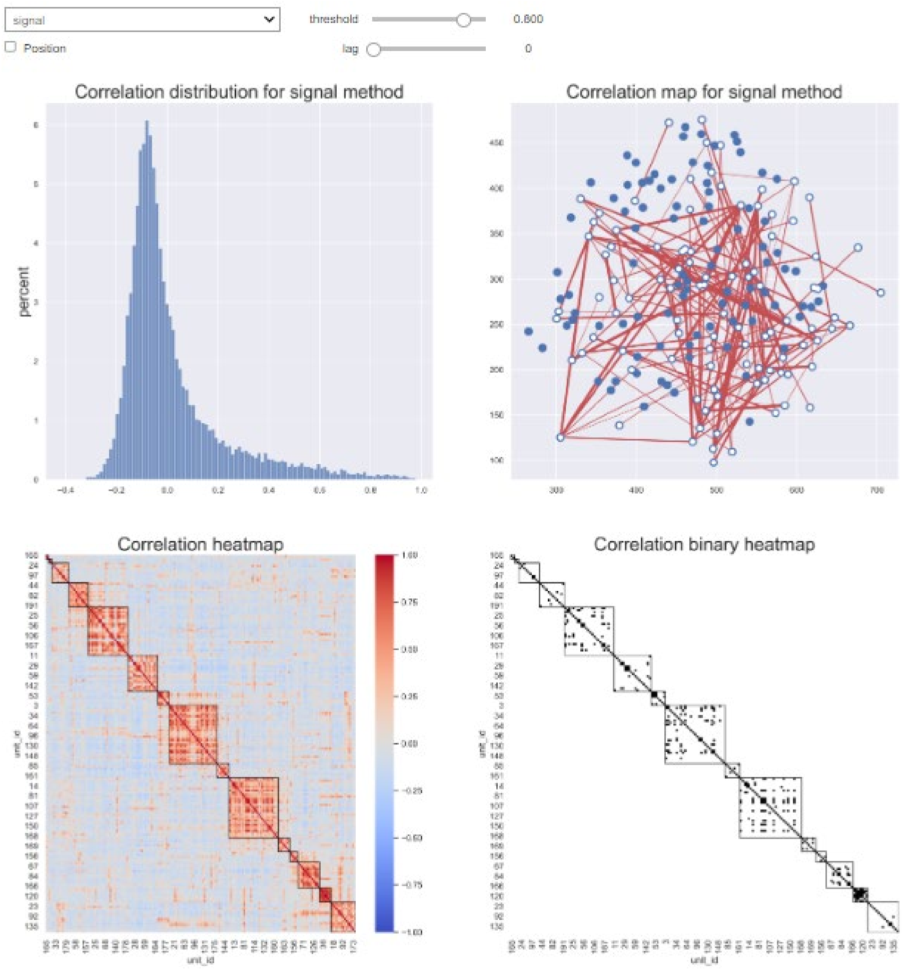
- Thresholds – a variable for individual preset threshold value of simultaneously active neurons.
- Save – saving the distribution in excel.



3.5. Correlation

- Choosing one of the possible types of interconnection between neurons:
    - signal - Pearson correlation coefficient for the intensity of the original signal,
    - diff – Pearson correlation coefficient by intensity derivative,
    - active – Pearson correlation coefficient based on binary results of active phase segmentation,
    - active_acc – the connection of intersection to the union of active states of neurons,
    - transfer_entropy – entropy of transfer between neurons.
- position – taking into account the spatial distance between neurons;
- threshold – threshold value for visualization;
- lag – maximum time shift of neurons relative to each other.
  maximum time shift of neurons relative to each other.

Automatically saved into "results".



signal

□ Position

threshold 0.800

lag 0

Correlation distribution for signal method

Correlation map for signal method

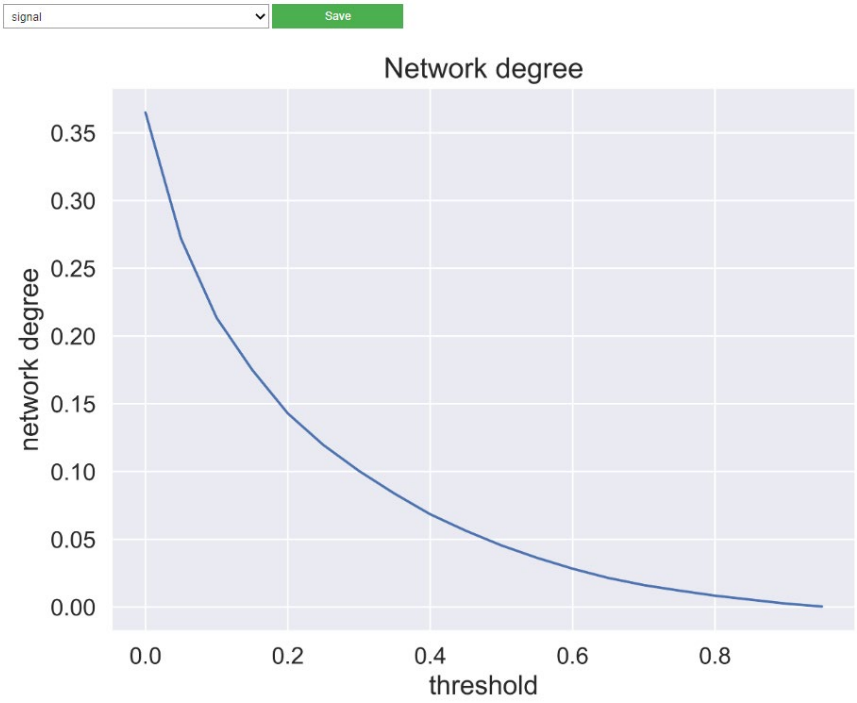
Correlation heatmap

Correlation binary heatmap

3.6. Network degree

Network degree – the proportion of connected neurons depending on the level determining the presence of a connection;

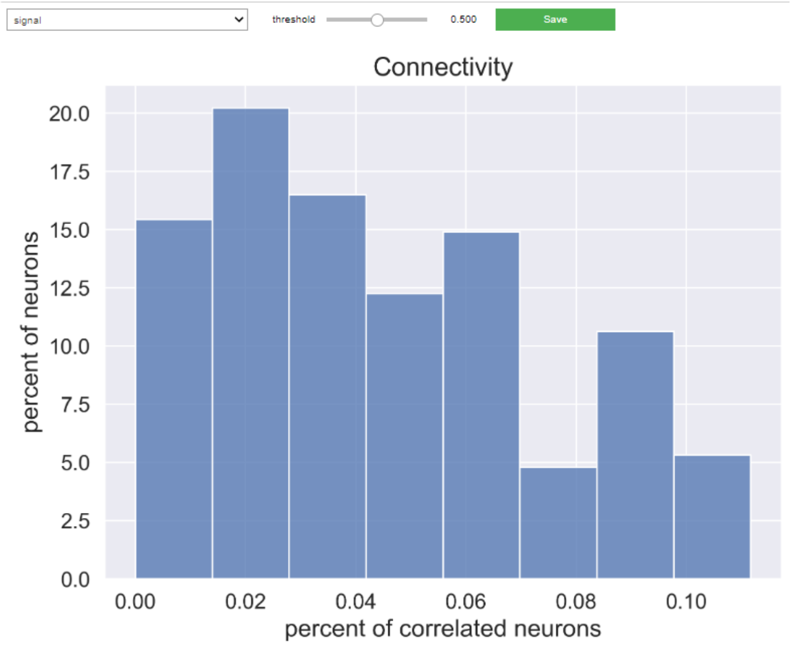
- Choosing one of the possible types of interconnection between neurons:
  - signal - Pearson correlation coefficient for the intensity of the original signal,
  - diff – Pearson correlation coefficient by intensity derivative,
  - active – Pearson correlation coefficient based on binary results of active phase segmentation,
  - active_acc – the connection of intersection to the union of active states of neurons,
  - transfer_entropy – entropy of transfer between neurons.
- Save – saving results as excel file.



3.7. Connectivity

Connectivity – distribution of the proportion of connectivity for each neuron.

- Choosing one of the possible types of interconnection between neurons:
  - signal - Pearson correlation coefficient for the intensity of the original signal,
  - diff – Pearson correlation coefficient by intensity derivative,
  - active – Pearson correlation coefficient based on binary results of active phase segmentation,
  - active_acc – the connection of intersection to the union of active states of neurons,
  - transfer_entropy – entropy of transfer between neurons.
- threshold – value for determination of the “strong” connection
- Save – saving results as excel file.

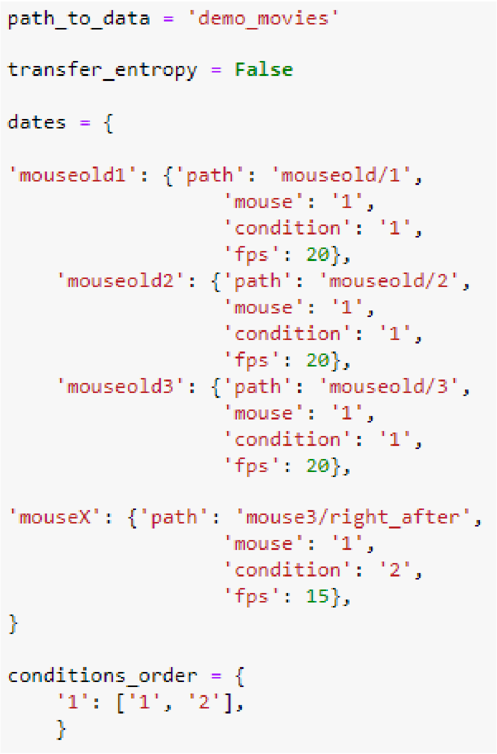


This Notebook **must be run first** of all because all the calculated statistics are used in the futher modules. All the results for each recording are saved into folder “results” in the same folder as it is stated in the “path_to_data”.

4. “Dimensionality reduction” module

4.1. Data

- path_to_data – the path to the root directory with the data
- transfer_entropy – using transfer entropy or not (when using much more time is needed for calculations);
- dates – information about all the recordings of interest:
  - key – unique record identificator (It must be unique for each mouse, as in example ‘mouseold1’, ‘mouseold2’ etc.),
  - values:
    - path – the path to the data relative to the root directory,
    - mouse – mouse identificator,
    - condition – condition of the mouse while recording,
    - fps – frames per second;
- conditions_order – chronological order for recordings;
  - key – unique identificator of the mouse (the same as in “dates”),
  - values – chronological order of recordings.

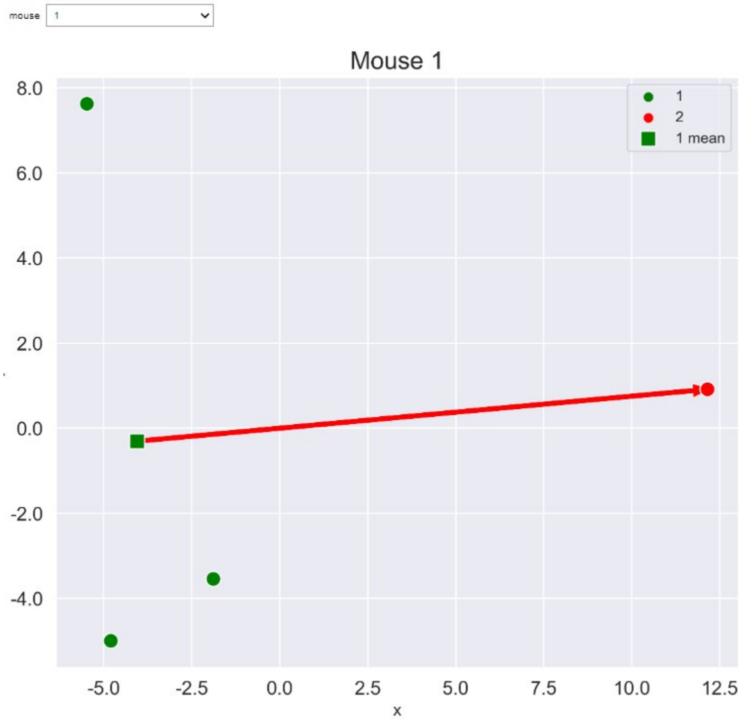


```
path_to_data = 'demo_movies'

transfer_entropy = False

dates = {

'mouseold1': {'path': 'mouseold/1',
                  'mouse': '1',
                  'condition': '1',
                  'fps': 20},
    'mouseold2': {'path': 'mouseold/2',
                  'mouse': '1',
                  'condition': '1',
                  'fps': 20},
    'mouseold3': {'path': 'mouseold/3',
                  'mouse': '1',
                  'condition': '1',
                  'fps': 20},

'mouseX': {'path': 'mouse3/right_after',
                  'mouse': '1',
                  'condition': '2',
                  'fps': 15},
}

conditions_order = {
    '1': ['1', '2'],
    }
```

4.2. PCA visualization

Visualization of results after PCA for each individual. The arrows visualize the chronological order specified in conditions_order.

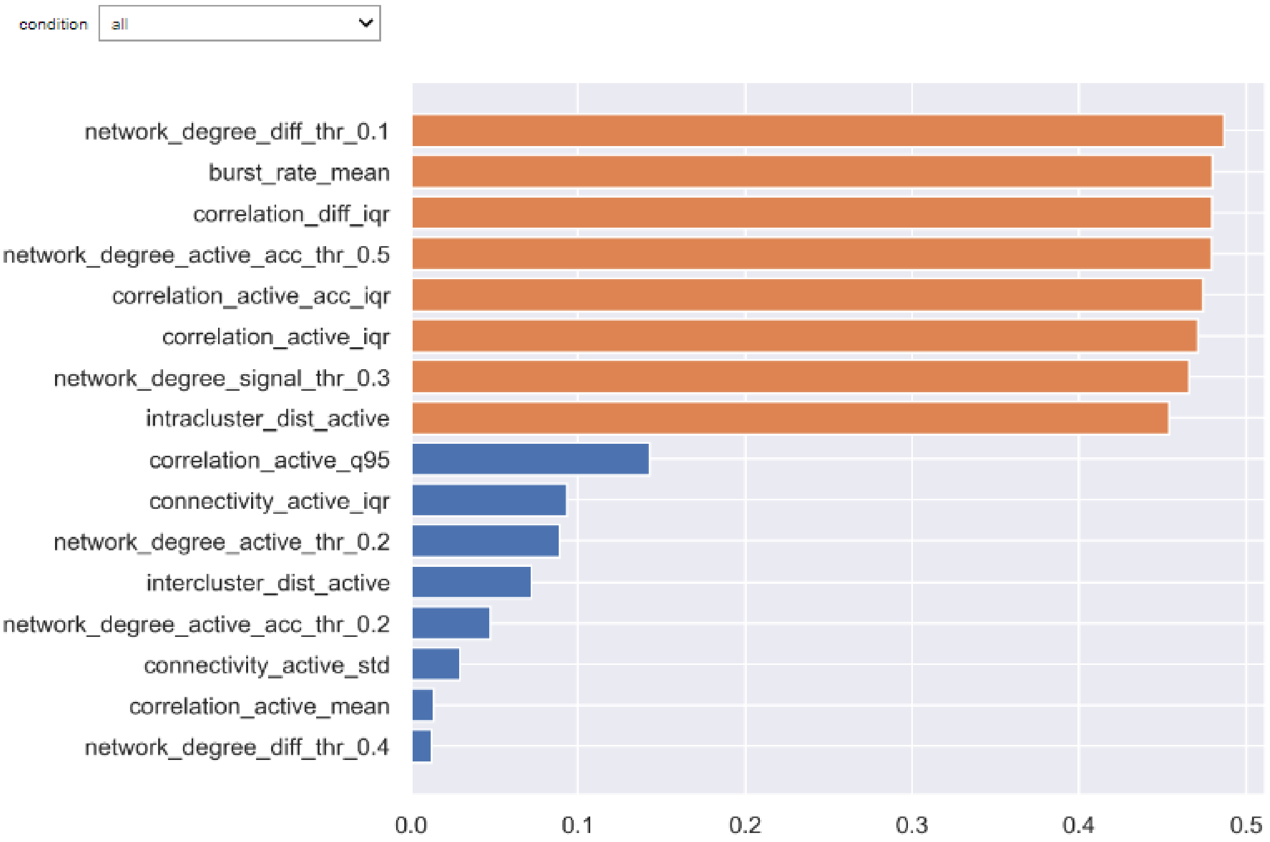
- mouse – mouse identificator to show;



Results of the PCA method are saved into folder “path_to_data”. Also, under this window at the step “8” coordinates for each point can be found.

Visualization of statistical deviation. The top 8 most highly deviant and stable statistics are displayed.

- condition – parameter to define whether visualization is made for all the conditions or only for chosen one.

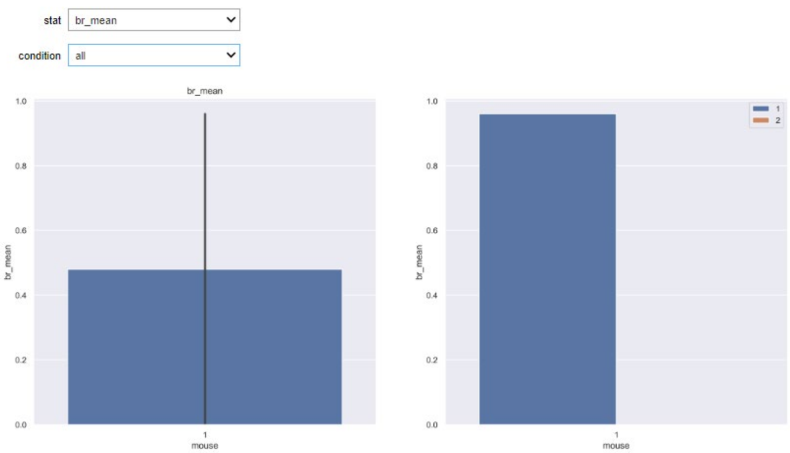


condition: all

4.3. Stats all

Visualization of the computed metrics. Here only 1 graph is presented due to in condition ‘2’ is only 1 recording.

- stat – setting the statistic of interest;
- condition – parameter to define whether visualization is made for all the conditions or only for chosen one.



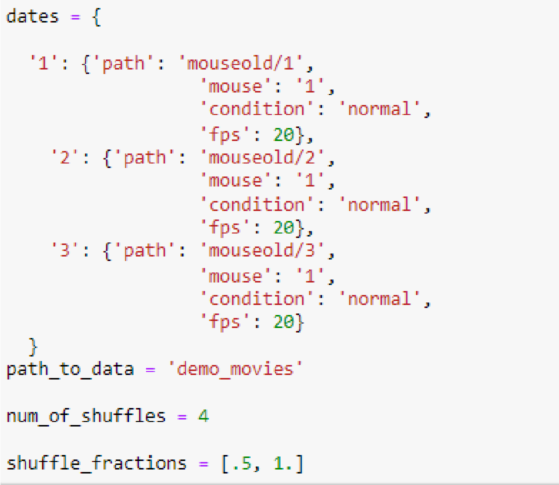
stat br_mean

condition all

5. Shuffling module (“MultipleShuffling”)

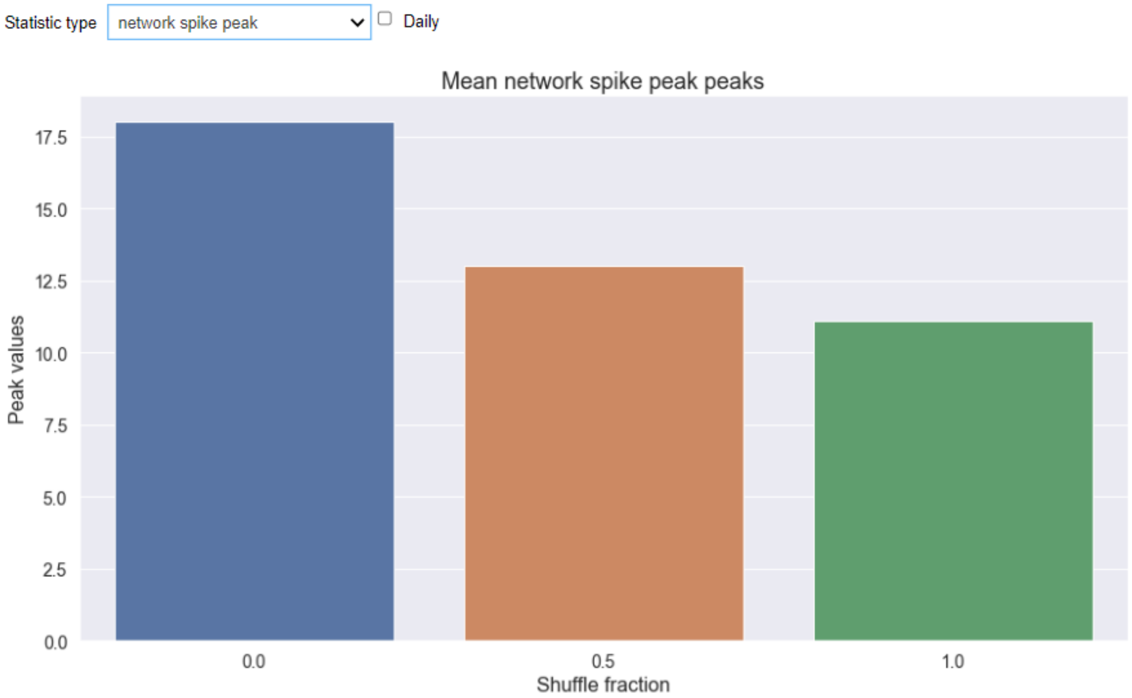
This notebook is created for shuffling the miniscope data to define level of variance and for comparison statistics from “original” data with “shuffled” one.

- path_to_data – the path to the root directory with the data;
- dates – information about all the recordings of interest:
  - key – title of the recording, should be unique;
  - values:
    - path – the path to the data relative to the root directory,
    - mouse – mouse identificator,
    - condition – condition of the mouse while recording,
    - fps – frames per second,
- num_of_shuffles – amount of the shuffling events. Usually for sufficient amount of recordings value of “1” is used. Can be increased for more robustness;
- shuffle_fractions – shuffle ratio for mixing module (can be manually changed).



```
dates = {

  '1': {'path': 'mouseold/1',
                'mouse': '1',
                'condition': 'normal',
                'fps': 20},
    '2': {'path': 'mouseold/2',
                'mouse': '1',
                'condition': 'normal',
                'fps': 20},
    '3': {'path': 'mouseold/3',
                'mouse': '1',
                'condition': 'normal',
                'fps': 20}
  }
path_to_data = 'demo_movies'

num_of_shuffles = 4

shuffle_fractions = [.5, 1.]
```

Being computed, there will be an option for choosing a metric of interest after shuffling with preset value of shuffle ratio. Shuffled data are saved into path_to_data folder as individual ".xlsx" tables.

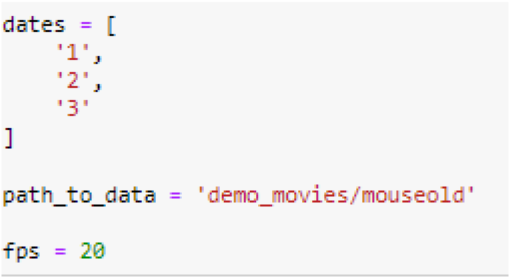


Statistic type: network spike peak ☐ Daily

6. Distance analysis

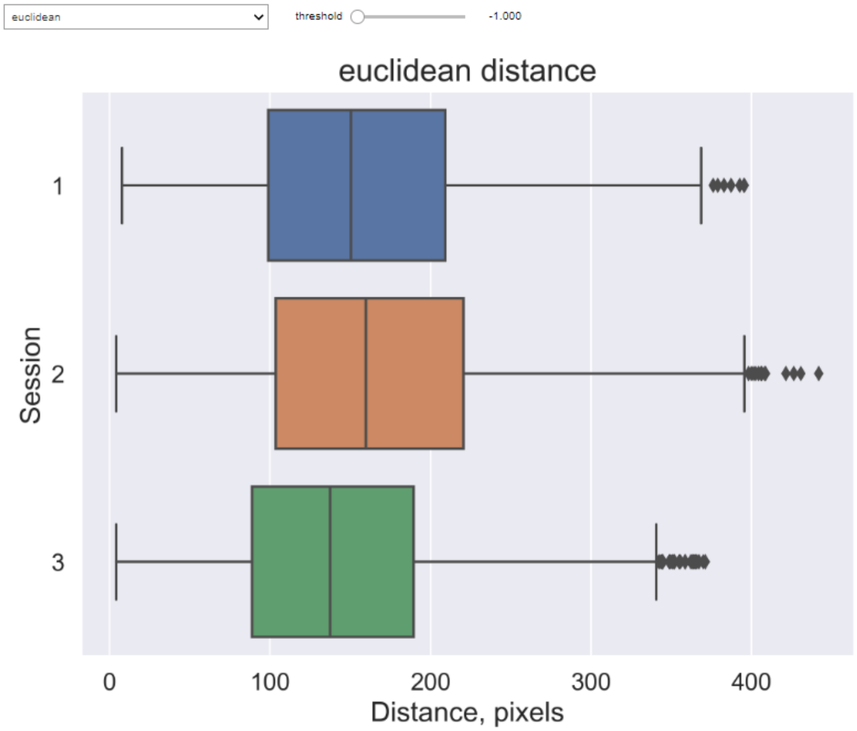
"Distance analysis" notebook is created for evaluating the distance criteria for pairwise correlations between neurons.

- path_to_data – the path to the root directory with the data;
- dates – information about all the recordings of interest:



```
dates = [
    '1',
    '2',
    '3'
]

path_to_data = 'demo_movies/mouseold'

fps = 20
```

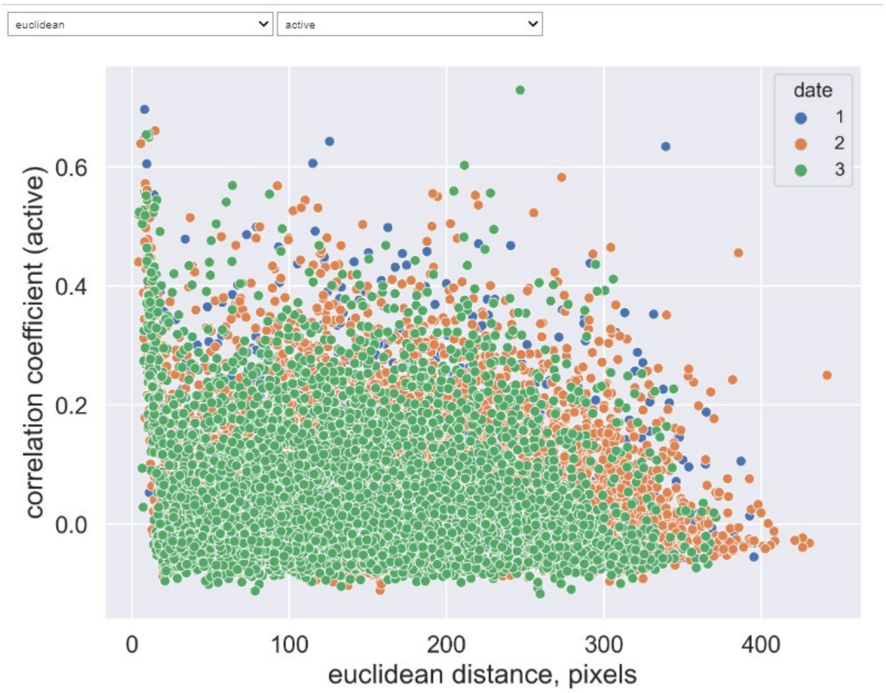
If the path stated correctly, graphs will be shown with possibility to save them in ".xlsx". For the step with calculating average distances between pairs of coactive neurons it is possible to choose between "Euclidian" and "radial" representation.



euclidean

threshold -1.000

Also by a widgets metrics of interest can be set for dependency visualization.



euclidean

active

All the presented metrics are or can be saved in the file with ".xlsx" format for further analysis, comparison and visualization.