– IntelliPy –

Installing and Running the Python GUI
for automatic Data Analyses
of IntelliCage data

INTELLIPY

PYTHON GUI FOR AUTOMATIC ANALYSES OF
INTELLICAGE DATA

Motivation:

The IntelliCage system helps researchers to conduct behavioral experiments and learning experiments with mice while ensuring minimal human intervention. The animals can be observed for long time periods - up to several weeks. This long-term data acquisition can provide new insights in mouse behavior, that might not be detectable in short-term observations. However, analyzing those big amounts of data is challenging for many researchers. IntelliPy aims to simplify and automize many aspects of the analysis, such as dividing data by learning module, acquiring data per group, creating learning curves or pivoting parameters in different timeframes. All plots are automatically created and the final tables for statistical tests are stored separately for the user.

Installation & Launch

...using pip

IntelliPy can be installed in principal for every operating system running Python. The easiest way, is installing it via the Python Packaging Index (PyPI) using the command:
pip install intellipy

It can then be used with
python3 -m intellipy or just **python -m intellipy**

...troubleshooting using Ubuntu

If you have an error due to the absence of the tkinter module in Ubuntu or Linux Mint, you can solve this issue with installing *python3-tk* by running
sudo apt-get update
sudo apt-get install python3-tk

in your terminal.

...using the intelli.exe

If you have problems installing IntelliPy using pip, you can also download a windows exe file here. As there is no need to have Python or any packages (e.g. pandas) installed, this file is quite big (75MB unzipped), as it brings Python and all recommended packages with it. Choosing the latter way for using IntelliPy, you just have to extract the *ZIP* file, open the directory and launch the *intelli.exe*.

Settings

IntelliPy utilizes the *Nosepoke.txt, Visit.txt* and *Animal.txt* file of an IntelliCage experiment. These files can be extracted using the TSE Analyzer. Additionally, if new groups should be assigned that are not included in the *Animal.txt* file, a group assignment file can be created, in order to conduct new group-wise analyses or to specify the alternative side conditions such as "Sucrose" or any other user-defined label for e.g. sucrose preference experiments:

Creating a group assignment file:

A tab-separated text file (tsv) can be used instead of the *Animal.txt* file for assigning new animal-group relationships. To address experiments with multiple conditions per animal, it is possible to relate an animal to more than one group. The group assignment file additionally contains information about the alternative label (if used). If such alternative labels were used in the IntelliCage experiment and are thus provided by the user in the group assignment file, the options when starting IntelliPy adjust, so that the user can choose, how to treat this alternative label. The group assignment file can be created as follows:

1. Give the word **Label** followed by a tab followed by the alternative label used in the experiment (if no alternative label was used, write *None* as alternative label.
2. Give the name of the first group followed by a tab and then all animals belonging to this group - also tab separated.
3. repeat step 2) for all remaining groups.

An example file is added as **"Group_assignment.txt"**

Conducted analyses:

Pivot tables:

For the parameters, measured by the IntelliCage Systems, such as **Lick Duration, Nosepoke Number** or **NosepokeDuration**, pivot tables are created for each module by IntelliPy. By default, these timeframes are created per day, but the user can add more timeframes using the IntelliPy GUI, like e.g. 12-hour or 6-hour timeframes. Depending on the chosen settings, group means or medians are calculated with their standard deviation or upper and lower quartiles respectively. For further statistical analyses, the pivoting results are stored in the resulting xlsx files as well as all created plots.

Learning Rates:

As the experiments conducted with the IntelliCage systems can be conducted as learning experiments with different setups per phase, the learning rate of each individual as well as for the each group can be of high interest. Rather than only the final rate of correct attempts, the rate per hour and per visit is computed and plotted by IntelliPy. This enables the user to utilize longitudinal learning information for each individual and group. For those learning rates, it is even possible to include all nosepokes or to remove those that were not followed by a lick. It can be argued about, whether a nosepoke without a lick should or shouldn't be accounted as a correct attempt, so this decision is up to the user. Furthermore, there are both, the possibility to exclude all nosepokes not followed by a lick or to treat them as incorrect attempts.

Alternative Label Analyses (e.g. Sucrose preference):

For learning experiments, including the choice between two sides after choosing the correct place, such as the choice between one side with Sucrose and one with Water, the proportion of Lick Duration spent for one side rather than the other side over time is computed by IntelliPy. Additionally, it is possible to define the alternative label (e.g. "Sucrose") as correct or as incorrect for the learning rate computation. This will influence the computation of the learning rate of the sucrose preference experiments. Consequently, for learning modules utilizing an alternative label, the nosepoke learning rate files are influenced by the used option and one additional file is created for showing the lick duration ratio between the "alternative label" side and the "Correct" side.

Output files

All output files are named by the following scheme:

<Module Number>_<Module Name>_<output suffix>

e.g. 1_FreeAdaptation_pivot_Data.xlsx

The following outputs are created:

- *nosepoke_learning_Data.xlsx*
- *pivot_Data.xlsx*
- *visit_learning_data.xlsx*
- *data.csv*

If alternative label is used, also:

- *learning_Data_Correct&Correct.xlsx*
- *alternative_Label_Data.xlsx*

These .xlsx output files contain the tabular data and also the plots, showing the data per individual and per group in two different plots per sheet.

nosepoke_learning_Data.xlsx

This file contains the table for the overall learning rate per individual and per group for nosepoke 1 - n in the first sheet. The second sheet contains the same information, but not computed for every single nosepoke but after each hour, so that it gives the possibility to compare the learning performance between individuals and groups based on time and not based on trials. Additionally, the given data is also visualized in line plots showing all individuals and all defined groups. If the user decided to treat nosepokes not followed by a lick differently (excluded or as incorrect), the data and plots are also created per nosepoke and per hour in different sheets. Additionally, the user can also decide to create sheets in which only nosepokes not followed by a lick are shown, e.g. if this behavior is assumed as playing behavior.

Both, the nosepoke.txt and the visit.txt file can be used for computing the learning rate but the latter contains information on the side error and the place error, whereas the former only contains information that are connected to nosepokes, so the place errors are not included in the data (as no nosepokes are possible at the wrong "place"). Additionally, the visit.txt file contains information per visit, not per nosepoke, so that many nosepokes can be included in one visit and thus temporal resolution is less detailed than in the *Nosepoke.txt* file. Furthermore, the *Visit.txt* data is not showing if a specific nosepoke was followed by a lick so that this part of the analysis is only included in the *nosepoke_learning_Data.xlsx*.

pivot_Data.xlsx

The aggregated data (sum) for each animal and the mean or median per group (depending on the settings) of the **LickDuration**, **LickNumber**, **NosepokeDuration**, **NosepokeNumber**, **VisitCount** and **LickContactTime** is given here as tables and graphs on different sheets. The graphs for the group's mean / median value also contain error bars either showing the standard deviation within the group or the 25th and 75th percentile. Per default, all data is aggregated by day. If the user specified some other hour intervals in the options, e.g. 12 hours, 6 hours, etc., these intervals will also be given here for each parameter in separate sheets.

visit_learning_Data.xlsx

This file contains the table for the overall learning rate per individual and per group for visit 1 - n in the first sheet. The second sheet contains the same information, but not computed for every single nosepoke but after each hour, so that it gives the possibility to compare the learning performance between individuals and groups based on time and not based on trials. Sheet number three and four contain the analogous information on the place errors, meaning information on the ratio of the individuals and groups choosing the correct corner of the cage. Additionally, the given data is also visualized in line plots showing all individuals and all defined groups.

data.csv

After all learning modules and their starting times are extracted automatically by IntelliPy, the original files for *Visit.txt* and *Nosepoke.txt* are sliced based on these modules. For each extracted module, the sliced version of the *Visit.txt* file is given here as a **csv** file in order to able to comprehend the exact data that was used for the analyses. IntelliPy also gives information in the *log.txt* file on whether the sum of all data in the sliced modules sum up to the original data.

Analyses with alternative label

learning_Data<alternative label>TreatedAsCorrect.xlsx_ or *learning_Data*<alternative label>TreatedAsIncorrect.xlsx_

Based on the chosen option, the alternative label is treated as correct or incorrect for the learning rate computation. Given this alteration, the file is computed just as the *nosepoke_learning_Data.xlsx*.

alternative_Label_Data.xlsx

This file shows the **LickDuration** ratio up to nosepoke/hour *x* of the alternative label per total **LickDuration**. For e.g. sucrose preference experiments, the **LickDuration** ratio of sucrose per total **LickDuration** per animal/group up to a given timepoint/nosepoke can be compared here.

The *log.txt* file

The *log.txt* file contains all parameters that were chosen by the user in order to ensure traceability of the results. Additionally, it is given, if the slicing of the original data into the different modules was successful by checking if the row numbers of all slices combined sum up to the row number of the original files. The path to the used input is given as well to make every part of the experiment traceable and thus reproducible.