Rethomics: an R framework to analyse high-throughput behavioural data

Quentin Geissmann[1*], Luis Garcia Rodriguez[2], Esteban J Beckwith[1], Giorgio F Gilestro[1*]

1 Department of Life Sciences, Imperial College London, London, United Kingdom
2 Affiliation Dept/Program/Center, Institution Name, City, State, Country

* qgeissmann@gmail.com, giorgio@gilest.ro

Abstract

Ethomics, a quantitative and high-throughput approach to ethology, is a new and exciting field. The recent development of methods that automatically score variables in multiple animals provides an unprecedented insight into the study of behaviours. The analysis of ethomics data presents many challenges that are conceptually independent of the acquisition platform. However, there is, little effort in providing a general framework to analyse multiple and long behavioural time series. We developed the `rethomics` framework, a suite of `R` packages that altogether provide utilities to: import, store, visualise and analyse behavioural data. The `rethomics` framework is available and documented at https://github.com/rethomics/.

Introduction

Animal behaviours are complex phenotypical manifestations of the interaction between nervous systems and their external or internal environment. In the last few decades, our ability to record large amounts of various phenotypical data has tremendously increased. Behaviour scoring is certainly not an exception to this trend. Indeed, many platforms (todo citations) have been developed in order to allow biologists to continuously record behaviours such as activity, position and feeding of multiple animals over long durations (days or weeks).

The availability large amounts of data is very exciting as it paves the way for in depth analyses. Clearly, the multiplicity of model organisms, hypotheses and paradigms makes the diversity of recording tools crucial. However, regarding subsequent data analysis, there is no unified, programmatic, framework that could be used as a set of building blocks in a pipeline. Instead, tools tend to consist of graphical interfaces with strictly defined functionalities that only import data from a single platform. There are three issues with this approach. First of all, state of the art analysis and visualisation requires a flexibility than only a programmatic interface can provide. Secondly, it favours replicated work as developers need to provide their own implementation to address very similar problems. Lastly, it links analysis and visualisation to the target platform, which makes it very difficult to share cross-platform tools.

Thankfully, behavioural data is conceptually largely agnostic of the acquisition platform and paradigm. Typically, the behaviour of each individual is a long time series (possibly multivariate and heterogeneous). In addition, each individual has to be unambiguously identified and associated with arbitrary metadata defined by the

experimenter (*e.g.* sex, treatment and genotype). Efficiently combining and manipulating these information, on datasets of hundreds of individuals, each recorded for weeks, is not trivial. The availability of a unified ethomics toolbox would help promoting the analysis of behaviour as a data science.

In the article herein, we describe `rethomics`, a framework that unifies analysis of behavioural dataset in an efficient and flexible manner. It offers an elegant solution to store, manipulate and visualise a large amount of data. Rethomics comes with a extensive documentation and a set of both practical and theoretical tutorials.

Design and Implementation

`rethomics` is implemented as a collection of small packages linked to one another (Fig **??**). This development model follows modern frameworks such as the `tidyverse`, which results in increased testability and maintainability. The different tasks of the analysis workflow (*i.e.* data import, manipulation and visualisation) are explicitly handled by different packages. At the core of `rethomics`, the `behavr` package offers a very flexible and efficient solution to store both large amounts data (*e.g.* position and activity) and metadata (*e.g.* treatment, genotype and so on) in a single `data.table`-derived object. Any input package will import experimental data as a `behavr` table which can, in turn, be manipulated and visualised regardless of the original input platform. Analyses results and plots integrate seamlessly within the `R` ecosystem, hence providing users with state-of-the-art visualisation and statistics tools.

Internal data structure

We created `behavr` (Fig **??**), a new data structure, based on the widely adopted `data.table` object, in order to address two challenges that inherent to manipulating ethomics results.

Firstly, there could be very long (typically $k_i > 10^8$), multivariate (often, $q > 10$), time series for each individual. For instance, each series could represent variables that encode coordinates, orientation, dimensions, activity, colour intensity and so on, sampled several times per second over days. Therefore, data structure must be computationally efficient – both in term of memory footprint and processing speed.

Secondly, a large amount of individuals are often studied (typically $n > 100$). Each individual (i) is associated with metadata, that is a set of p "metavariables" that describe experimental conditions. For instance, metadata stores information regarding the date and location of the experiment, treatment, genotype, sex, post-hock observations and other arbitrary metavariables. It is good practice to record as many metavariables as possible so they can later be used as covariates. Therefore, typically $p > 10$.

`behavr` tables link metadata and data within the same object, extending the syntax of `data.table` to access, and manipulate metadata. This approach guaranties that any data point can be mapped correctly to its parent metadata. It also allows implicit update of metadata when data is altered. For instance, when is data filtered, only the remaining individuals should be in the new metadata. Metadata and data also have to interoperate. For instance, when one wants to update variable according to the value of a metavariable (say, alter the variable x only for animals with the metavariable $sex =$ "$male$").

Data import

Data import package translate results from a recording platform (e.g. text files and databases) into a single `behavr` object. Currently, we provide a package to read Drosophila Activity Monitor (DAM2) data and another one for Ethoscope data. Although the structure of the raw results if very different, conceptually, loading data is very similar. In all cases the user is asked to generate a metadata table (one row per individual). The columns are of two types: mandatory and optional. The mandatory ones are the necessary and sufficient information to fetch data (*e.g.* machine id, region of interest and date). The optional columns are user-defined arbitrary fields that relate to the experiment itself (*e.g.* condition and sex).

In this respect the metadata file is standardised an comprehensive data frame describing of an experiment. Using such a structure comes with multiple advantages. For instance, it simplify collaboration and data exchange as all treatments and individuals are very explicit. Then, it promotes good experimental practices such as interspersion of treatments (indeed, without it, users are tempted to simplify their design, for instance, confounding device/location and treatment). Furthermore, it streamlines the inclusion and analysis of further replicates in the same workflow. Indeed, additional replicates can simply be added as new rows, and replicate number later user, if needed, as a covariate.

Visualisation

Long time series often need to be preprocessed before visualisation. Typically, users are interested in understanding the individual or population trends over time. To integrate visualisation in `rethomics`, we implemented `ggetho`, a package extending the widely adopted `ggplot2` by providing preprocessing tools as well as new layers and scales. Our tools make full use of the internal `behavr` structure to deliver efficient representations of temporal trends. It particularly applies to the visualisation of long experiments, with the ability to, for instance, annotate light and dark phases, wrap time over a circadian day, display "double-plotted actograms" and periodograms. Importantly, `ggetho` is fully compatible with `ggplot`.

Circadian and sleep analysis

The packages `zeitgebr` and `sleepr` provide tools to analyse circadian behaviours and sleep, respectively. Together, they offer a suite of methods to compute periodograms and define find peaks, score sleep from inactivity (*e.g.* using the "five minute rule"), and characterise the architecture of sleep bouts (*e.g.* number, length and latency).

Results

We present an small example examining the circadian behaviour of 64 fruit flies in recorded in a DAM2 – a paradigm very widely adopted. A less formal description of this case and others are explained at https://rethomics.github.io/. In order to prodive a more comprehensive and didactic example, data was altered and simplified. Fig **??** describes our case experiment (A) and the corresponding metadata (B). Briefly, three monitors were used each containing 16 males and 16 females of a different genotype (A, B and C). Two replicates were performed at different times. raw data and metadata files are publicly available TODO cite zenodo

Data loading

Quality control

Actograms

Spectogram

Availability and Future Directions

All packages in the `rethomics` framework are available under the terms of the GPLv3 license and listed at https://github.com/rethomics/. Extensive installation instructions as well as reproducible demos and tutorials are available at https://rethomics.github.io/. All packages are continuously integrated and unit tested on several version of `R` to minimise resent and future issues.

Acknowledgements

TODO:
Han Kim
Maite
Patrick Krätschmer

References

1. Conant GC, Wolfe KH. Turning a hobby into a job: how duplicated genes find new functions. Nat Rev Genet. 2008 Dec;9(12):938–950.

2. Ohno S. Evolution by gene duplication. London: George Alien & Unwin Ltd. Berlin, Heidelberg and New York: Springer-Verlag.; 1970.

3. Magwire MM, Bayer F, Webster CL, Cao C, Jiggins FM. Successive increases in the resistance of Drosophila to viral infection through a transposon insertion followed by a Duplication. PLoS Genet. 2011 Oct;7(10):e1002337.