aion: An R Package to Represent Archaeological Time Series

Nicolas Frerebeau

1 UMR 6034 Archéosciences Bordeaux, Maison de l’Archéologie, Université Bordeaux Montaigne, 33607 Pessac cedex, France

Summary

aion is designed to provide a consistent framework for representing archaeological time series that can extend very far in the past. aion provides a system of classes and methods to represent and work with such time series. This package does not provide tools for temporal analysis or modeling. Instead, it offers a system of classes and methods to represent and work with archaeological time series. This API can be extended and used by other specialized packages (see kairos v2.0 as an example).

Statement of need

R ships with a lot of functionality useful for time series, in particular in the base stats (R Core Team, 2023), or in the zoo (Zeileis & Grothendieck, 2005) packages. However, these features are not adapted to most archaeological time series. At the same time, numerous R packages have been developed to describe, analyze, and model temporal data in the context of archaeological studies in the broadest sense. These packages encompass various functionalities, including handling radiocarbon data (e.g., Bchron by Haslett & Parnell, 2008; rcarbon by Crema & Bevan, 2021), Optically Stimulated Luminescence dating (Luminescence by Kreutzer et al., 2012), Bayesian chronological modeling (ArchaeoPhases by Philippe & Vibet, 2020), using paleoenvironmental proxies (e.g., shoredate by Roalkvam, 2023), or other temporal data (e.g., kairos by Frerebeau, 2023). This multitude of packages underscores the significance of computational approaches in archaeology (Schmidt & Marwick, 2020). However, it also presents a major challenge as each package employs its own representation of temporal information. Consequently, exchanging data between different packages within the same data workflow becomes even more arduous.

Archaeological data is typically collected through field excavations or surveys, resulting in irregularly spaced observation times. Although several packages can handle irregular time series, the way they represent dates means they cannot easily be used for archaeological series. These are indeed defined for a given calendar era and more importantly they can involve dates very far in the past.

Functionality

In base R, dates are represented by default as the number of days since 1970-01-01 (Gregorian), with negative values for earlier dates. aion uses a different approach: it allows to create date vectors represented as rata die (Reingold & Dershowitz, 2018), i.e. as number of days since

---

1See the CRAN Task View about time series analysis: https://cran.r-project.org/view=TimeSeries.
01-01-01 (Gregorian). This allows to represent dates independently of any calendar and makes calculations and comparisons easier.

The rata die vector provides the internal time representation of the aion time series (note that the era (Roe, 2022) package allows to work with numeric vectors that represent year-based time scales). The fixed() function allows to create such a vector from dates that can then be converted back into dates (or years) of a particular calendar.

In aion a time series is represented by an S4 class that inherits from the base array. A time series object can be created with the series() function that returns an $n \times m \times p$ array, with $n$ being the number of observations, $m$ being the number of series and with the $p$ columns of the third dimension containing extra variables for each series. This array comes with an extra time slot that store the observations times expressed in rata die. It be created with the series() function.

All output produced by aion can be formatted with (virtually) any calendar, as long as the calendar has been defined and the associated conversion methods are available. aion natively supports both Julian and Gregorian calendars (with the most common eras for the latter, e.g. Before 2000, Before Present, (Before) Common Era…) and allows to create custom calendars.

Acknowledgements

The author would like to thank Brice Lebrun for creating the package logo. Joe Roe also deserves a special mention for the development of the era (Roe, 2022) package, which inspired the present work.

References


