

# GeoChronR – an R framework to model and analyze age-uncertain paleogeoscientific timeseries.

Nicholas McKay<sup>1</sup>, Julien Emile-Geay<sup>2</sup>, and Deborah Khider<sup>2</sup>

<sup>1</sup>School of Earth and Sustainability, Northern Arizona University, Flagstaff, AZ 86011

<sup>2</sup>University of Southern California, Los Angeles, CA

**Correspondence:** Nicholas McKay (Nicholas.McKay@nau.edu)

**Abstract.** Chronological uncertainty is a hallmark of the paleosciences. While many tools have been made available to researchers to produce age models suitable for various settings and assumptions, disparate tools and output formats often discourage integrative approaches. In addition, associated tasks like propagating age model uncertainties to subsequent analyses, and visualizing the results, have received comparatively little attention in the literature and available software. Here we describe GeoChronR, an open-source R package to facilitate these tasks. GeoChronR is built around emerging data standards for the paleosciences (Linked Paleo Data, or LiPD), and offers access to four popular age modeling techniques (Bacon, BChron, Oxcal, BAM). The output of these models is easily stored in LiPD, enabling paleo-aware synchronization, regression, correlation, principal component, and spectral analyses. Five real-world use cases illustrate how to use GeoChronR to facilitate these tasks, and to visualize the results in intuitive ways.

*Copyright statement.* The author's copyright for this publication is transferred to institution/company.

## 1 Introduction

### 1.1 Background

Quantifying chronological uncertainties, and how they might influence the understanding of past changes is fundamental to the paleogeosciences. Without robust error determination, it is impossible to properly assess the extent to which past changes occurred simultaneously across regions or the duration of abrupt events, both of which limit our capacity to apply paleoscientific understanding to modern and future processes. The need for better infrastructure to both characterize uncertainty, and to explicitly evaluate how age uncertainty impacts the the interpretation of records of past climate, ecology or landscapes, has been long recognized (?, (more)). In response to this need, the paleogeoscience community has made substantial advances toward improving geochronological accuracy by:

1. Improving analytical techniques that allow for more precise age determination on smaller and context-specific samples (?, ?, ?, ?, ?)

2. Refining our understanding of how past changes in the Earth system impact the age accuracy, for example: improvements to the radiocarbon calibration curve (e.g., Stuiver & Reimer, 1993; Reimer et al., 2004) and advances in our understanding of spatial variability in cosmogenic production rates used in exposure dating (e.g., Brown et al., 2001; Brown & Hodges, 2004).
3. Dramatic improvement in the level of sophistication and realism in age-depth models used to estimate the ages of sequences between dated samples (e.g., Blatter, 1986; Blatter & Haeberli, 1987).

Over the past 20 years, these advances have been widely adopted in the paleogeosciences. However, despite the progress made in quantifying uncertainty in ages and in age models, few studies have formally evaluated how chronological uncertainty may have affected their results. For instance, whereas the algorithms presented by (e.g., Blatter, 1986; Blatter & Haeberli, 1987) have been broadly used, the overwhelming majority of these studies calculate the single best-estimate model (often a median or mean), use this model to put measured paleoclimatic or paleoenvironmental data on a timescale, and then proceed to analyze the record with little to no reference to the age modeling exercise (e.g., Blatter, 1986; Blatter & Haeberli, 1987). Typically, any discussions of chronological uncertainties remain qualitative.

This paradigm is beginning to change. In recent years a handful of studies have taken advantage of approaches that generate ensembles of age models to evaluate how the results of their analyses and conclusions vary given differences between ensemble members (e.g., Blatter, 1986; Blatter & Haeberli, 1987). By using each ensemble age model to create a time-uncertain ensemble records, and then carrying that ensemble through the analysis, the precise impact of age uncertainty can be formally evaluated. This approach, of course, does not address all aspects of uncertainty, but it does offer the broad potential to ascertain which results are robust to chronological uncertainty, and which are not.

Despite its potential to substantially improve uncertainty quantification for the paleogeosciences, this framework is not widely utilized. The majority of studies utilizing this approach have been regional [e.g., Blatter, 1986; Blatter & Haeberli, 1987] or global-scale (e.g., Blatter, 1986; Blatter & Haeberli, 1987) syntheses. Occasionally, primary publications of new records incorporate time-uncertain analysis into their studies (e.g., Blatter, 1986; Blatter & Haeberli, 1987), but this remains rare. We suggest that there are several reasons for the lack of adoption of these techniques:

1. For synthesis studies, the necessary geochronological data are not publicly available for the vast majority of records. Even when they are available, the data are archived in diverse and unstructured data formats. Together, this makes what should be a simple process of aggregating and preparing data for analysis prohibitively time-consuming;
2. For studies of new and individual records, few tools for ensemble analysis are available, and those that are require a degree of comfort with coding languages and scientific programming that is rare among paleogeoscientists;
3. There is a disconnect between age-model development and time-uncertain analysis. Published approaches have utilized either simplified age-modeling approaches (e.g., Blatter, 1986; Blatter & Haeberli, 1987), or specialized approaches not used elsewhere in the community (e.g., Blatter, 1986; Blatter & Haeberli, 1987).

Extracting the relevant data from commonly-used age-modelling algorithms, creating time-uncertain ensembles, then reformatting those data for analysis in available tools typically requires the development of extensive custom codes. An integrative approach is needed to facilitate this work.

## 1.2 Motivation

GeoChronR was built to lower the barriers to broader adoption of these emerging methods. It provides an easily-accessibly, open-source and extensible software package of industry-standard and cutting-edge tools that provides users with a single environment to create, analyse, and visualize time-uncertain data. GeoChronR is designed around emerging standards in the paleogeosciences that connects users to growing libraries of standardized datasets formatted in the Linked PaleoData format (?) and (?).

## 1.3 Outline of manuscript

This manuscript describes the design, analytical underpinnings and most common use cases of GeoChronR. Section ?? describes the integration of age modelling algorithms with GeoChronR. Section ?? details the methods implemented for age uncertain analysis. Section ?? goes through the principles and implementation of age-uncertain data visualization in GeoChronR, and section ?? provides five real-world examples of how GeoChronR can be used for paleoscientific workflows.

## 2 Age Uncertainty Quantification in GeoChronR

GeoChronR does not introduce any new approaches to age uncertainty quantification; rather, it integrates existing, widely-used packages while streamlining the acquisition of age ensemble members. Fundamentally, there are two types of age models used in the paleogeosciences: tie-point and layer-counted. Most of the effort in age uncertainty quantification in the community has been focused on tie-point modelling, where the goal is to estimate ages (and their uncertainties) along a depth profile given chronological estimates (and their uncertainties) at multiple depths downcore. Over the past 20 years, these algorithms have progressed from linear or polynomial regressions with simple characterizations of uncertainty (?, ?) to more rigorous techniques, particularly Bayesian approaches: as of writing, the three most widely used algorithms are Bacon (?), BChron (?), and OxCal (?), which are all Bayesian age-deposition models that estimate posterior distributions on age-depth relationships using different assumptions and methodologies. ? reviewed the performance of these three algorithms, as well as a non-Bayesian approach (?), and found that the three Bayesian approaches generally outperform previous algorithms, especially when appropriate parameters are chosen (although choosing appropriate parameters can be challenging). Bacon, BChron and Oxcal all leverage Monte Carlo Markov Chain (MCMC) techniques to sample the posterior distributions, thereby quantifying age uncertainties as a function of depth in the section. GeoChronR interfaces with each of these algorithms through their R packages (?, ?) (cite other R Packages), streamlining input and output. %data input and the extraction of the age ensembles from the MCMC results for further analysis.

In addition to working with ensembles from tie-point age models, GeoChronR connects users to probabilistic models of layer-counted chronologies. BAM (?) was designed to probabilistically simulate counting uncertainty in banded archives, such as corals, ice cores, or varved sediments, but can be used to crudely simulate age uncertainty for any record, and is useful when

the data or metadata required to calculate an age-depth model are unavailable. Here we briefly describe the theoretical basis and applications of each of the four approaches integrated in GeoChronR.

## 2.1 Bacon

The Bayesian ACcumulatiON (Bacon) algorithm (?) is one of the most broadly used age-modelling techniques, and was designed to take advantage of prior knowledge about the distribution and autocorrelation structure of sedimentation rates in a sequence to better quantify uncertainty between dated levels. The algorithm employs an adaptive Markov Chain Monte Carlo algorithm that allows for Bayesian learning to update the sedimentation rate distribution. Bacon has two key parameters: the shape of the accumulation prior, and the segment length, which can interact in complicated ways (?). In our experience, the segment length parameter has the greatest impact on the ultimate shape and amount of uncertainty simulated by Bacon, as larger segments result in increased flexibility of the age-depth curve, and increased uncertainty between dated levels. Bacon is written in C++ and R, with an R interface. More recently, the authors released an R package `rbacon` (?), which GeoChronR leverages to provide access to the algorithm. Bacon will optionally return a subset of the MCMC accumulation rate ensemble members with high *a posteriori* probabilities, which GeoChronR uses to form age ensemble members for subsequent analysis.

depositional model?

## 2.2 BChron

BChron (?, ?) uses a similar approach, using a continuous Markov monotone stochastic process coupled to a piecewise linear deposition model. This simplicity allows semi-analytical solutions that make BChron computationally efficient. BChron was originally intended to model radiocarbon-based age-depth models in lake sedimentary cores of primarily Holocene age, but its design allows broader applications. In particular, modeling accumulation as additive independent gamma increments is appealing for the representation of hiatuses, particularly for speleothem records, where accumulation rate can vary quite abruptly between quiescent intervals of near-constant accumulation (?, ?,?). The downside of this assumption is that BChron is known to exaggerate age uncertainties in cases where sedimentation varies smoothly (?).

Adjustable parameters? {Deborah}

## 2.3 Oxcal

The OxCal software package has a long history and extensive tools for the statistical treatment of radiocarbon and other geochronological data (?). In ?, age-depth modelling was introduced with three options for modelling depositional processes that are typically useful for sedimentary sequences: uniform, varve, and Poisson deposition models, labeled U-sequence, V-sequence and P-sequence, respectively. The Poisson-based model is the most broadly applicable for sedimentary, or other accumulation-based archives (e.g. speleothems), and although any sequence type can be used in GeoChronR, most users will use a P-sequence, which is the default. Analogously to segment length parameter in Bacon, the  $k$  parameter (called “eventsPerUnitLength” in GeoChronR), controls how many events are simulated per unit of depth, and has a strong impact on the

flexibility of the model, as well as the amplitude of the resulting uncertainty. As the number of events increases, the flexibility of the model, and the uncertainties decrease. ? found that this parameter has large impact on the accuracy of the model, and moreso than the choices made in Bacon or Behron. Fortunately, ? made it possible for  $k$  to be treated as a variable, and the model will estimate the most likely values of  $k$  given a prior estimate and the data, however this calculation can greatly increase the convergence time of the model. Oxcal is written in C++, with an interface in R (?). Oxcal doesn't calculate posterior ensembles for a depth sequence, but can optionally output MCMC posteriors at specified levels in the sequence. GeoChronR uses this feature to extract ensemble members for subsequent analysis.

## 2.4 Banded Age Model (BAM)

? is a probabilistic model of age errors in layer-counted chronologies. The model allows a flexible parametric representation of such errors (either as Poisson or Bernoulli processes), and separately considers the possibility of double-counting or missing a band. The model is parameterized in terms of the error rates associated with each event, which are intuitive parameters to paleogeoscientists, and may be estimated via replication (?). In cases where such rates can be estimated from the data alone, an optimization principle may be used to identify a more likely age model when a high-frequency common signal can be used as a clock (?). As of now, BAM does not consider uncertainties about such parameters, representing a weakness of the method. Bayesian generalizations have been proposed (?), which could one day be incorporated into GeoChronR if the code is made public.

BAM was coded in MATLAB and R, and it is this latter version that GeoChronR uses.

## 3 Age-uncertain data analysis in GeoChronR

Some theoretical work has attempted to quantify how chronological uncertainty may affect various paleoscientific inferences (e.g. ?); however, such an approach is necessarily piecemeal, and therefore impractical given the variety of age-uncertainty structures in real-world paleogeoscientific data. Consequently, GeoChronR follows a pragmatic and broadly-used approach that leverages age ensembles, and optionally ensembles in climate proxy or paleoenvironmental data, to propagate uncertainties through all steps of an analysis. Effectively, this is done by randomly sampling from the ensemble(s) and then repeating the analysis on each sample (typically hundreds to thousands) to build an output ensemble that quantifies the impact of those uncertainties on a particular inference. These output ensembles often do not lend themselves to binary significance statistics (e.g., a p-value below), but are readily used to provide quantiles that estimate probability density ranges, and can provide strong evidence for which results are robust to age and proxy uncertainty (and which are not). Version 1.0.0 of GeoChronR has implemented ensemble analytical techniques for four of the most common analyses in the paleogeosciences: correlation, regression, spectral and principal component analyses.

### 3.1 Correlation

Pearson’s product-moment correlation is the most common measure of association between variables.

Its computation is fast, lending itself to ensemble analysis, with a handful of pretreatment and significance considerations that are relevant for ensembles of paleogeoscientific data.

First, correlation analysis for timeseries is built on the assumption the datasets can be aligned on a common timeline. Age-uncertain data violate this assumption. We overcome this by treating each ensemble member from one or more age uncertain timeseries as valid for that iteration, then “bin” each of the timeseries into coeval intervals. The “binning” procedure in GeoChronR sets up an interval, which is typical evenly spaced, over which the data are averaged. Generally, this intentionally degrades the median resolution of the timeseries, for example, a timeseries with 37-year median spacing could be reasonably “binned” into 100- or 200-year bins. The binning procedure is repeated for each ensemble member, meaning that between different ensembles, different observations will be placed in different bins.

Following binning, Pearson correlation is calculated and recorded for each ensemble member. In addition to the correlation coefficients, GeoChronR calculates their significance for each correlation. Paleogeoscientific timeseries are often highly auto-correlated, which can lead to spurious assessments of significance (?). Therefore, in addition to the standard Student’s T-test p-value calculation, we also calculate a p-value that is adjusted for autocorrelation to reflect the reduction in degrees of freedom due to autocorrelation following ?. However, repeating this test over multiple ensembles raises the issue of test multiplicity (?), or “look elsewhere effect”. To overcome this problem, we control for this false discovery rate using the simple approach of ?, coded in R by ?.

### 3.2 Regression

Linear regression is a commonly used tool to model the relationships between paleogeoscientific data and instrumental or other datasets. One application is calibration-in-time [references], whereby a proxy timeseries is calibrated to an instrumental series with a linear regression model over their period of overlap. This approach is particularly vulnerable to age uncertainties, as both the development of the relationship, and the reconstruction, are affected. GeoChronR propagates age (and optionally proxy) uncertainties through both the fitting of the ordinary least squares regression model, and the reconstruction “forecast” using the ensemble model results and age uncertainty. Like the correlation algorithm, ensemble regression uses an ensemble binning procedure that’s analgous to correlation. GeoChronR then exports uncertainty structure of the modeled parameters (e.g. slope and intercept), as well as the ensemble of reconstructed calibrated data through time.

### 3.3 Principal Component Analysis

GeoChronR implements the age-uncertain principal component analysis (PCA) procedure introduced by ?, with some minor modifications and additions. Like correlation and regression, PCA (or empirical orthogonal function {EOF} analysis) requires temporally aligned observations, and GeoChronR uses a binning procedure to achieve this across multiple ensembles. This differs from the implementation of ?, who interpolated the data to a common timestep. In addition, traditional singular value

decomposition approaches to PCA require a complete set of observations without any missing values. For paleoclimate data, especially when considering age uncertainty, this requirement is often prohibitive. To overcome this, GeoChronR implements multiple options for PCA analysis using the `pcaMethods` package. The default and most rigorously tested option is a probabilistic PCA (PPCA) approach that uses expectation maximization algorithms to infill missing values (?). This algorithm assumes that the data and their uncertainties are normally distributed, which is often (but not always) a reasonable assumption for paleogeoscientific data. As in correlation and regression, GeoChronR propagates uncertainties through the analysis by repeating the analysis across randomly sampled age and/or proxy ensemble members to build output ensembles of the loadings (eigenvectors), variance explained (eigenvalues) and principal component timeseries. Because the sign of the loadings in PCA analyses is arbitrary and vulnerable to small changes in the input data, GeoChronR reorients the sign of the loadings for all PCs so that the mean of the loadings is positive. For well defined modes this effectively orients ensemble PCs, but loading orientation may be uncertain for lower order, or more uncertain, modes.

### 3.4 Spectral Analysis

Many research questions in the paleosciences revolve around spectral analysis: describing phase leads and lags among different climate system components over the Pleistocene (SPECMAP, ?), the hunt for astronomical cycles over the Holocene (?, ?) or in deep time (?, ?,?), or characterizing the continuum of climate variability (?, ?). Yet, spectral analysis in the paleosciences faces unique challenges: chronological uncertainties, of course, as well as uneven sampling, which violates the assumptions of classical spectral methods (?).

To facilitate the quantification of chronological uncertainties in such assessments, GeoChronR implements 4 spectral approaches: 1. the Lomb-Scargle periodogram (?), which uses an inverse approach to harmonic analysis in unevenly-spaced timeseries.

2. REDFIT, a version of the Lomb-Scargle periodogram tailored to paleoclimatic data (?, ?, ?). The R implementation of REDFIT is inherited from the `dplR` package (?). 3. the wavelet-based method of ?, called `nuspectral`. This method is quite similar to the Weighted Wavelet Z-transform algorithm of ?, though it is prohibitively slow in this implementation, and the fast version using a compact-support approximations of the mother wavelet did not perform well in our tests. 4. The multi-taper method (MTM) of ?, a mainstay of spectral analysis (?) designed for evenly spaced timeseries. GeoChronR re-uses its ? implementation, which couples MTM to efficient linear interpolation, together with various utilities to define autoregressive and power-law benchmarks for spectral peaks.

## 4 Visualization with GeoChronR

One of the challenges with age-uncertain analysis is that it adds at least one additional dimension to the results, which can be challenging to visualize. GeoChronR aims to facilitate simple creation of intuitive, publication-quality figures that provide multiple options for visualizing the impacts of age-uncertainty, while maintaining flexibility for users to customize their results as needed. To meet the multiple constraints of simplicity, quality and customization, GeoChronR relies heavily on the

ggplot2 package (?). High-level plotting functions in GeoChronR (e.g., `plotTimeseriesEnsRibbons` and `plotPca`) produce complete figures as ggplot2 objects, that can be readily customized by adding or changing ggplot2 layers. The figures in the Use Cases section (Section @ (sec:use-cases)) are all produced directly from GeoChronR without modification and generally fall into a few categories.

{Light Mode vs Dark Mode}

## 4.1 Timeseries

Most of the figures that GeoChronR produces are ensemble timeseries. GeoChronR uses two complementary approaches to visualize these ensembles. The first is the simplest, where a large subset of the ensemble members are plotted as semi-transparent lines. This approach, implemented in `plotTimeseriesEnsLines`, provides a faithful representation of the data, while the overlapping semi transparency provides a qualitative sense of the ensemble uncertainty structure. The second approach uses contours to more rigorously visualize the structure of the time-value uncertainty space represented by the ensembles. `plotTimeseriesEnsRibbons` shows the quantiles of the ensembles at specified levels as shaded bands. This approach provides the quantitative uncertainty structure, but tends to smooth out the apparent temporal evolution of the data. Fortunately, the two approaches are complementary, and often the best approach is to quantify the ensemble distribution with ribbons in the background, and then overlap them with a handful of ensemble lines to illustrate the structure in representative ensemble members.

## 4.2 Maps

GeoChronR has simple mapping capabilities built in that rely on the `maps` (?) and `ggmap` (?) packages. The `mapLipd` and `mapTs` functions provide quick geospatial visualization of one or more datasets, but also serve as the basis for the visualization of ensemble spatial data produced by ensemble PCA analyses. In paleogeoscientific studies, the loadings (eigenvectors) of a PCA analysis are often portrayed as dots on a map, with a colorscale that highlights the sign and amplitude of the loadings. In ensemble PCA, the additional dimension of uncertainty in the loadings needs to be visualized as well. In GeoChronR, the median of the loadings is shown as a color, and the size of the symbol is inversely proportional to the spread of uncertainty across the ensemble. Consequently, large symbols depict loadings that are robust to the uncertainties, whereas small symbols show datasets whose loadings change substantially across the analysis. An example is shown in Section @ (sec:pca)

## 4.3 Spectra

It is customary to plot spectra on a log-log scale, which helps separate close-by frequencies. This choice also naturally highlights scaling laws (?, ?) as linear structures in this reference frame. GeoChronR thus implements this convention. In addition, the abscissa ( $\log_{10} f$ ) is labeled according to the corresponding period, which are more intuitive than frequency to scientists reading the plot. To help identify significant periodicities, confidence limits can be superimposed, based on user-specified benchmarks (see ??). The `plotSpectrum()` function visualizes single ensemble members (e.g. a median age model), while



`plotSpectraEns()` visualizes the quantiles of a distribution of age-uncertain spectra as ribbons, using the eponymous `ggplot` function.

## 5 Use cases

We now illustrate the use of these tools on five use cases. The first example shows how to create age ensembles on different archives, and how to visualize the timing of abrupt events with appropriate uncertainty quantification. The second example shows how to quantify similarities in age-uncertain records. The third introduces the topic of age-uncertain calibrations, the fourth quantifies multivariate relationships using principal components analyses, and the fifth deal with spectral analysis.

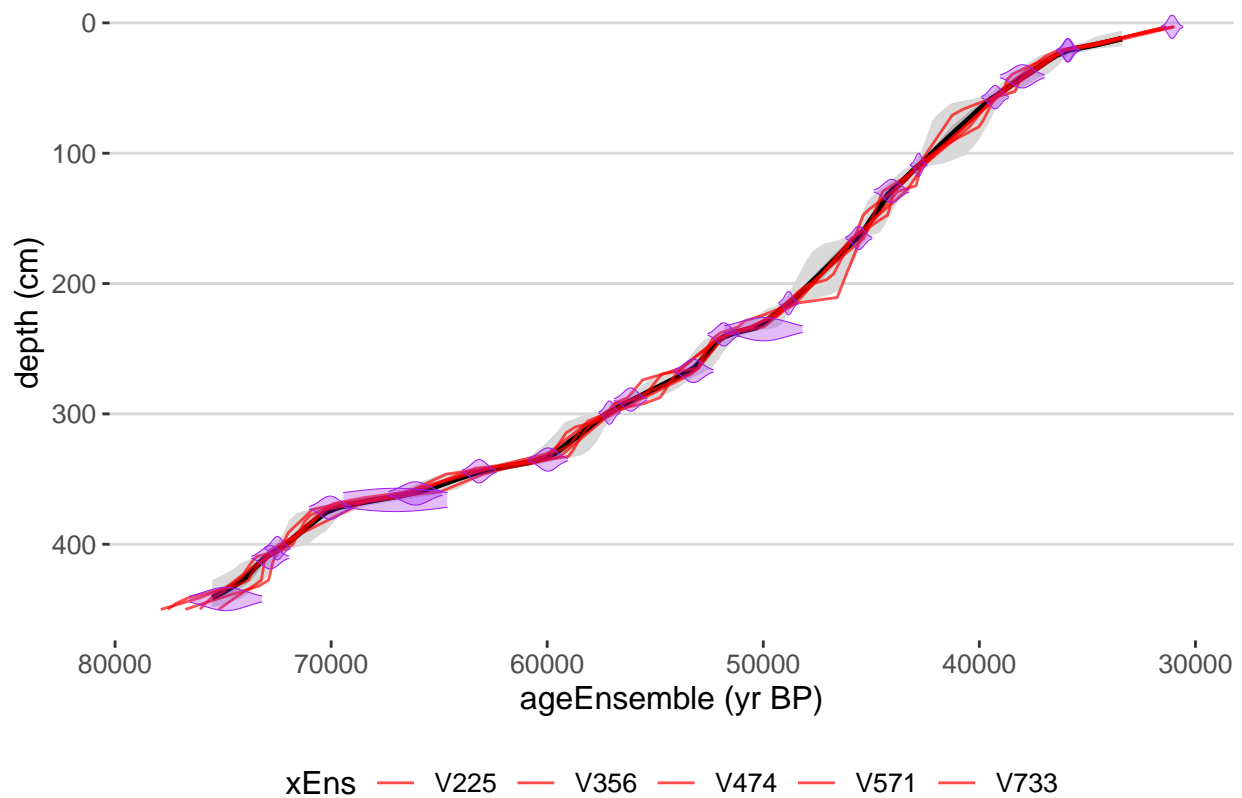
### 5.1 Creating an age ensemble

A common first task when using `geoChronR` is to create an age ensemble, either because the user is developing a new record, or because the age ensemble data for the record they are interested is unavailable. As described in section X.Y, workflows for four published age quantification programs are integrated into `geoChronR`. All four methods are simply used in `geoChronR` with a `LiPD` file that contains the chronological measurements, and the functions `runBacon(L)`, `runBchron(L)`, `runOxcal(L)` and `runBam(L)`. These functions take `LiPD` objects as inputs, and return updated `LiPD` objects that include age-ensemble data generated by the respective software packages, with these data stored in appropriate tables. Typically, additional parameters are needed for to optimally run the algorithms. When these parameters are not included, `geoChronR` will run in interactive mode, asking the user which variables and parameters they would like to model. These parameter choices are printed to the screen during while the program runs, or are available later with the function `getLastVarString()`. By specifying these parameters, age model creation can be scripted and will run in non-interactive mode. In this use case, we'll use `geoChronR` and `BChron(?)` to calculate an age ensemble for the Hulu Cave  $\delta^{18}\text{O}$  speleothem record (?), and `BAM(?)` to simulate age uncertainties for the GISP2 ice core  $\delta^{18}\text{O}$  dataset (?). The `plotChronEns(hulu)` function will plot an age-depth model and uncertainties derived from the age ensemble.

After an age ensemble has been added to a `LiPD` object, the user can visualize the ensemble timeseries using `plotTimeseriesEnsRi` and `plotTimeseriesEnsLines()`. GISP2  $\delta^{18}\text{O}$  is plotted with age uncertainty, using both functions, in figure x.

### 5.2 Correlation

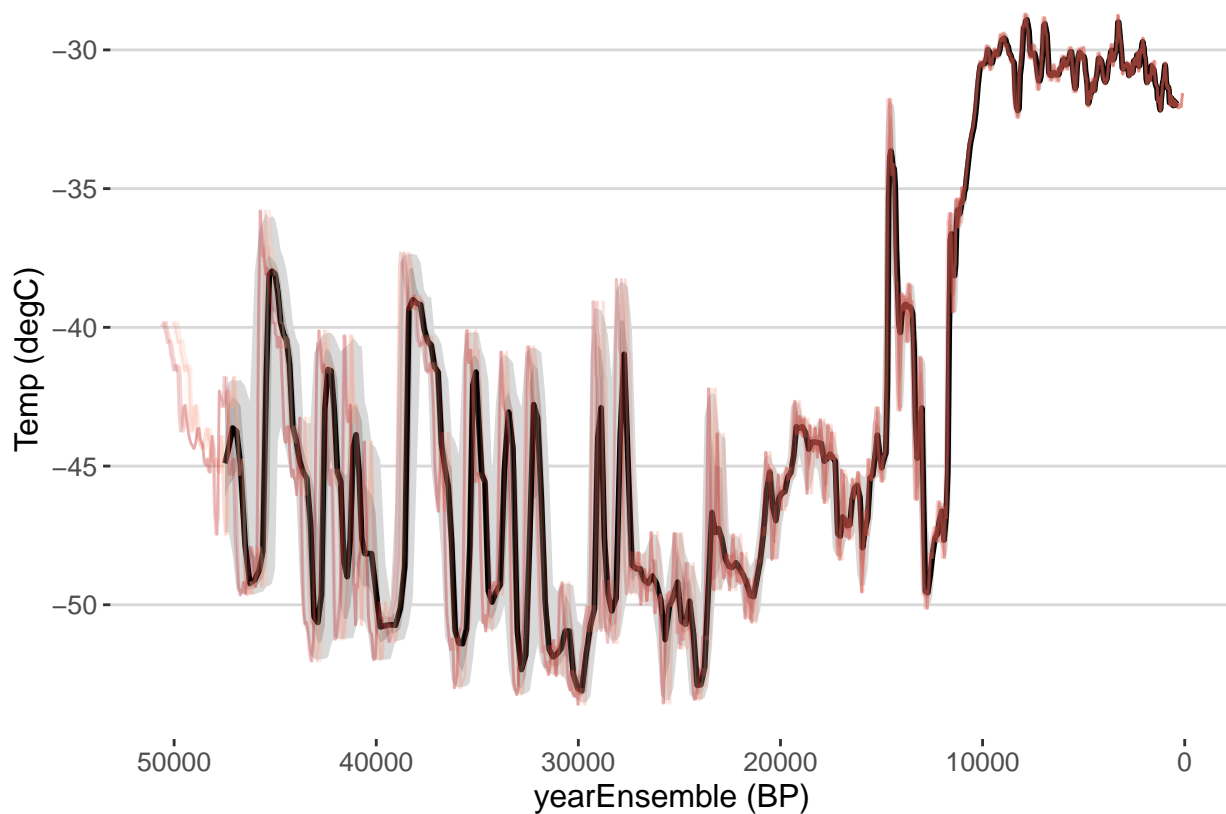
Now that the user has generated age ensembles for the two datasets, they're interested to see if a correlation between the two datasets is robust to the age uncertainty modeled here. On multi-millennial timescales, the two datasets display similar features, and previous work has suggested that abrupt events of the Last Glacial period, which are observed in the GISP2 record, can impact the Asian Monsoon and be observed in speleothem records such as the Hulu Cave dataset. (NM: add references here and flesh out background) The `corEns()` function in `geoChronR` will calculate ensemble correlations across age-uncertain datasets, such as these. `corEns()` will also sample across ensembles in the `paleoData` as well, if present. Here we calculate



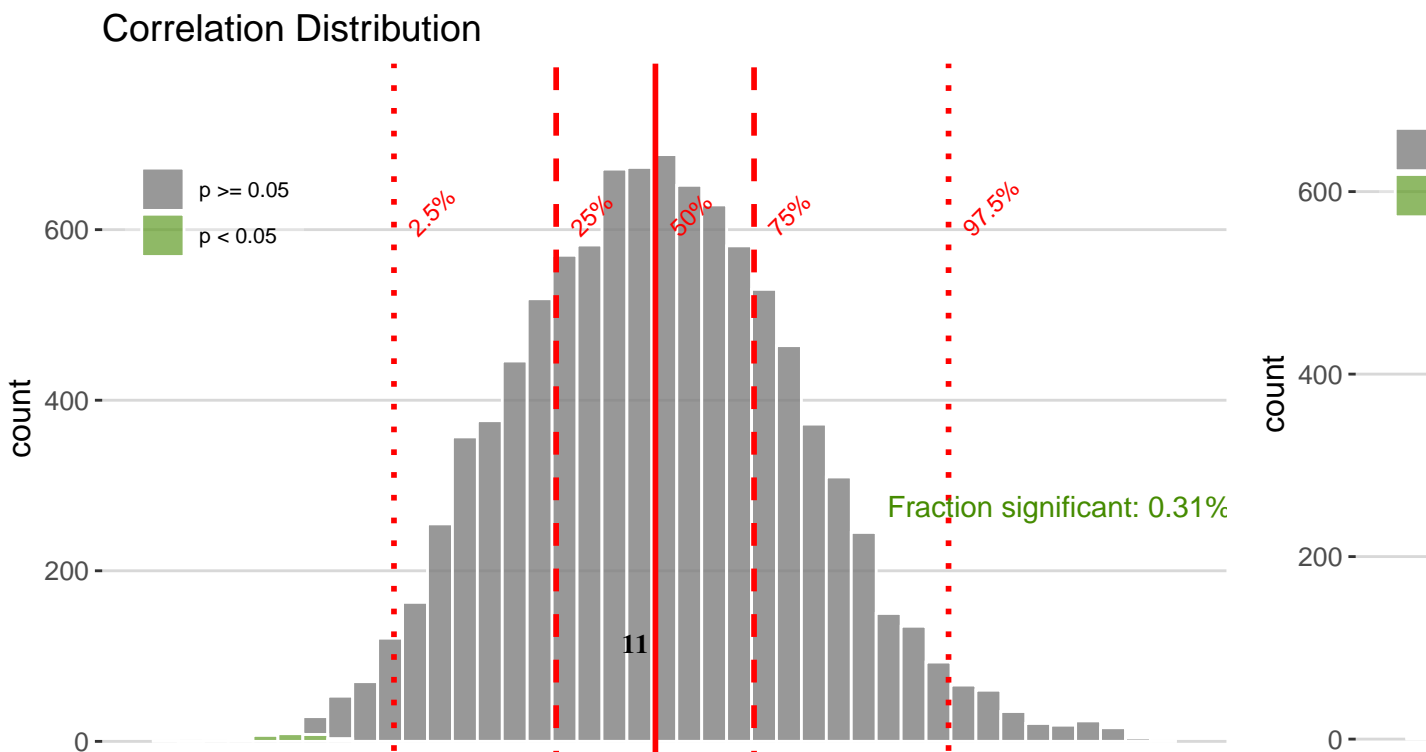
**Figure 1.** Need to write a caption

correlations during the period of overlap in 500 yr steps, determining significance for each pair of ensemble members while accounting for autocorrelation.

The results show consistently negative correlations, although 17.72% of the ensemble members are positive. However, only 0.31% are significant after accounting for serial autocorrelation.



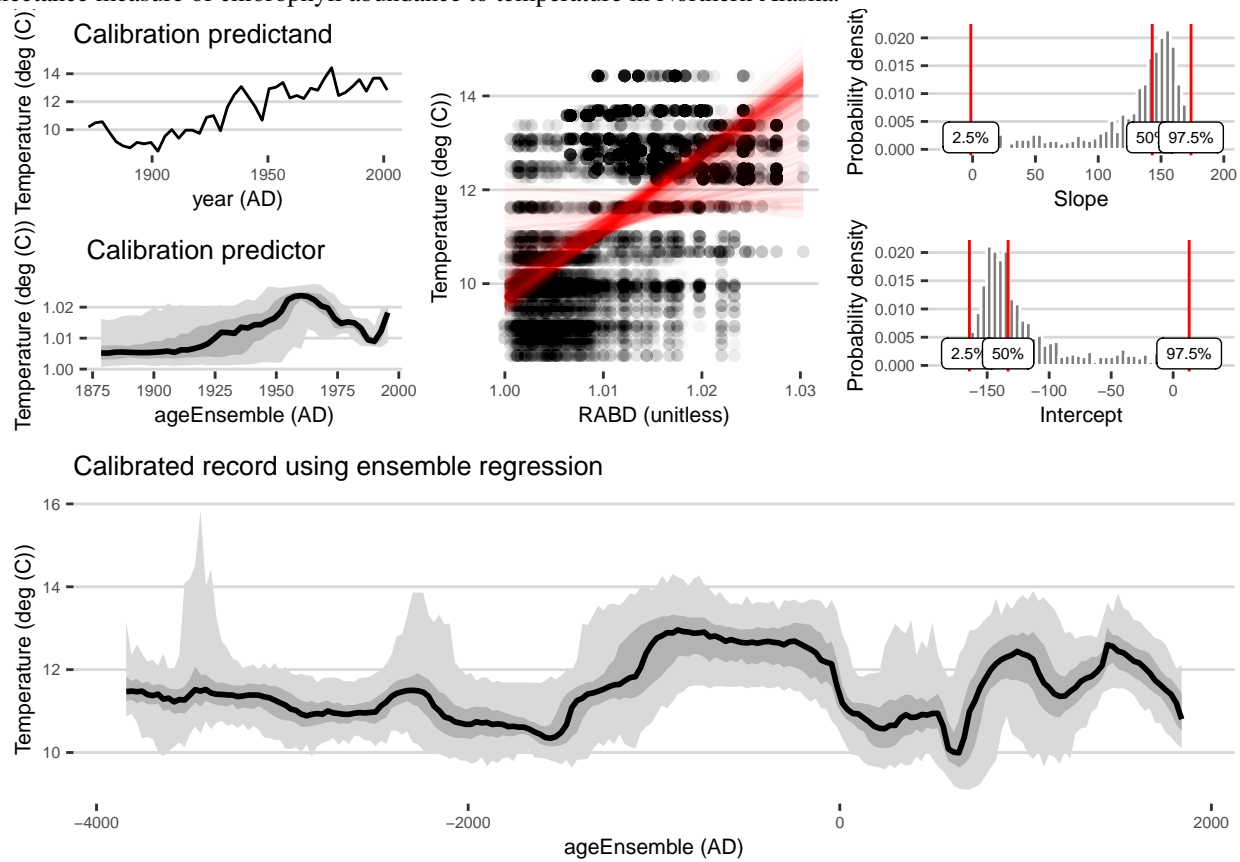
**Figure 2.** Need to write a caption



In this use case, we demonstrate how a user could calculate and visualize and age-uncertain correlation between the Hulu speleothem  $\delta^{18}\text{O}$  record and the GISP2 ice core  $\delta^{18}\text{O}$  record. (Introduction about why a user might want to do an age uncertain correlation between Hulu and GISP2).

5.3 Age-uncertain Calibration

A natural extension of ensemble correlation is ensemble regression. Although there are use cases where regressing one age-uncertain variable onto another is called for, here we regress an age-uncertain paleoclimate proxy onto time-certain instrumental to develop a calibration-in-time. For this use case, we reproduce the results of ?, where the authors calibrate a spectral reflectance measure of chlorophyll abundance to temperature in Northern Alaska.

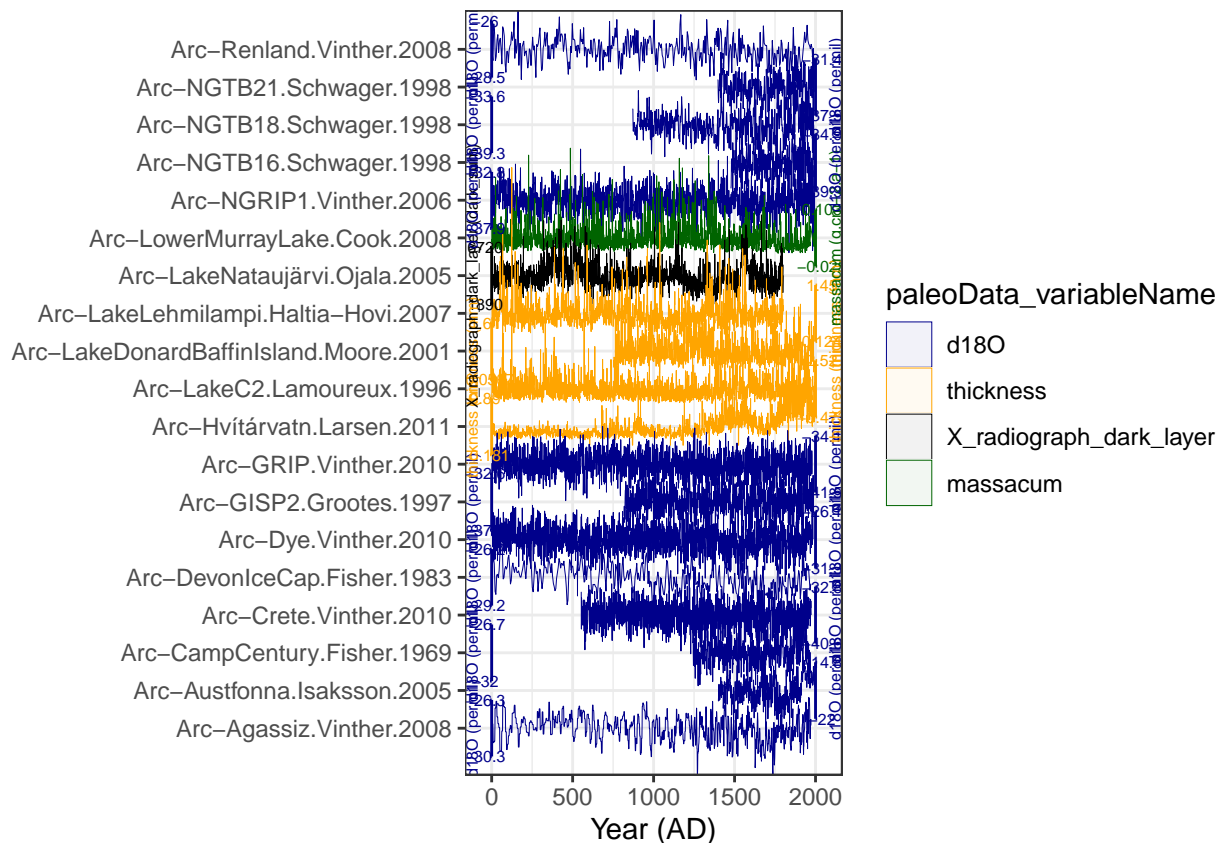


Figure

captions.

5.4 Principle Component Analysis {sec:pca}

Thus far, the use cases have highlighted age-uncertain analyses of one or two sites, however quantifying the effects of age uncertainty can be even more impactful over large spatial datasets. Here we showcase how to use geoChronR to perform



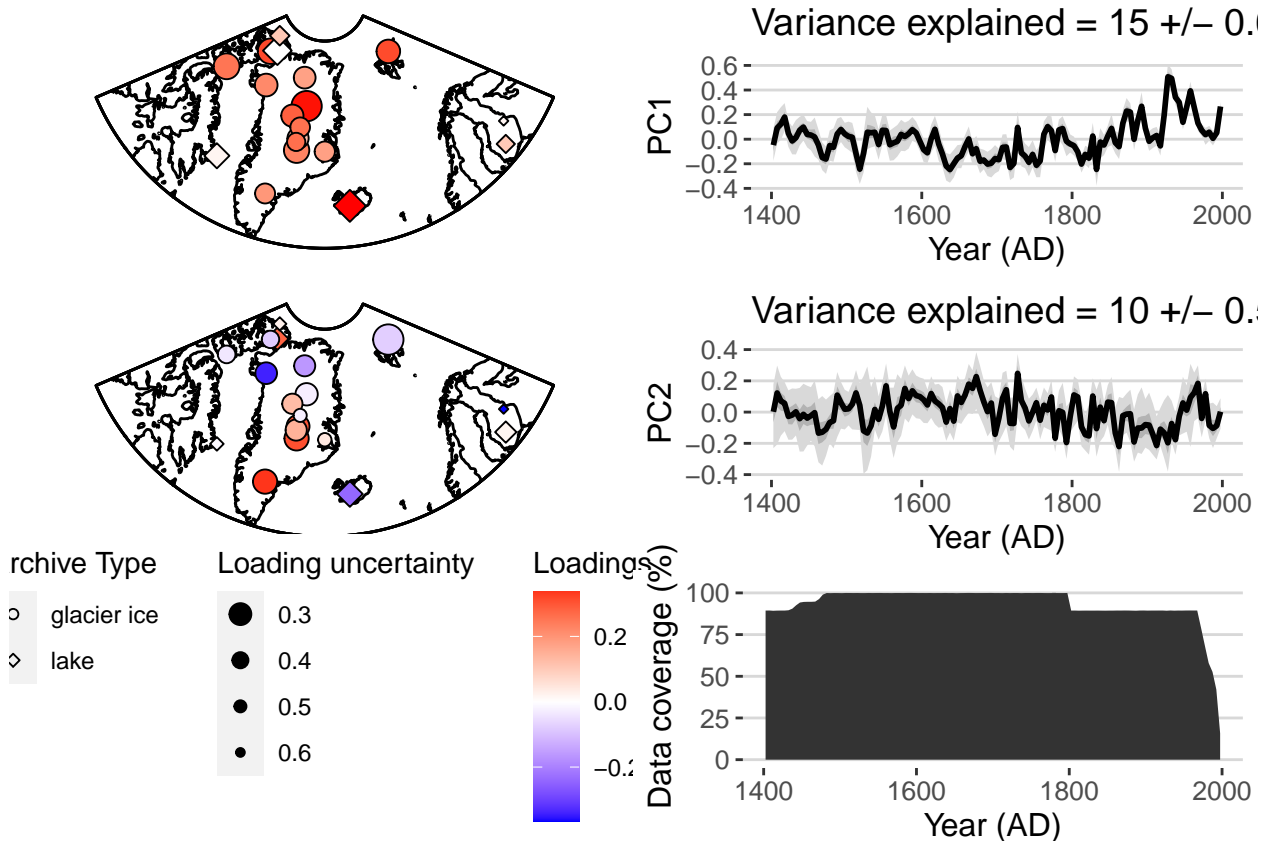
**Figure 3.** Need to write a caption

age-uncertain principle components analysis (PCA), also known as Monte Carlo Empirical Orthogonal Function (MCEOF) analysis, pioneered by ?. In this example, we examine the Arctic 2k database (?).

## 5.5 Spectral Analysis

To illustrate the use of spectral analysis functionalities, consider the following use case: a paleoceanographer wants to identify the relative energy of oscillations at orbital (Milankovitch) periodicities in a deep-sea sediment core, and quantify the impact of age uncertainties on this assessment. To do so, they select the benthic  $\delta^{18}\text{O}$  record of the International Ocean Drilling Project core 846 (?), covering the past 4.7 million years. For this assessment, we use a new age model obtained by alignment to the benthic  $\delta^{18}\text{O}$  stack of ? using the HMM-Match algorithm (?). HMM-Match is a probabilistic method that generates an ensemble of 1000 possible age models compatible with the chronostratigraphic constraints; this ensemble was archived as a table in the associated LiPD file.

Now let us take a first look at the median age model:



**Figure 4.** Need to write a caption

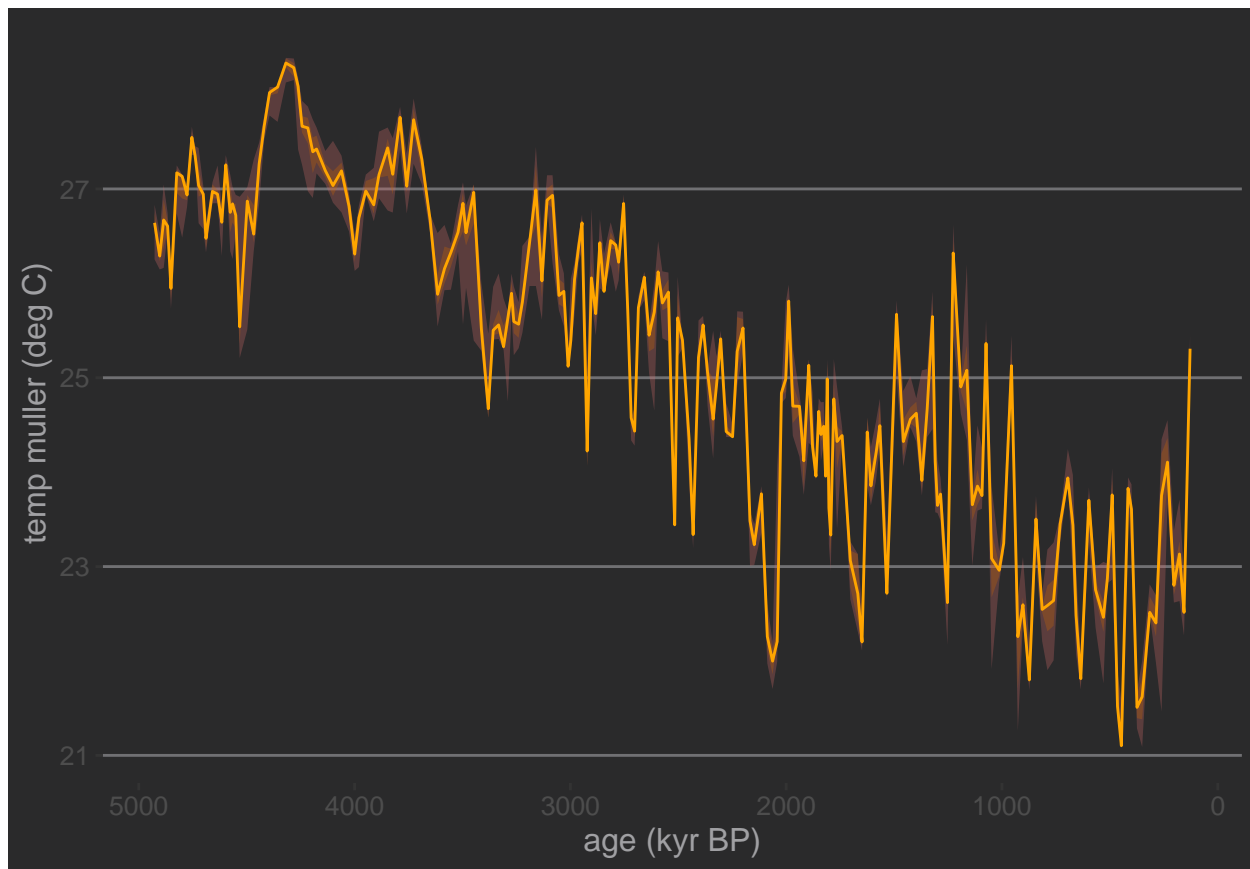
This paleoclimate record displays three salient features:

- a long-term cooling trend ( $\delta^{18}\text{O}$  gets more positive over time) characteristic of the late Neogene and Quaternary climate.
- quasi-periodic oscillations (the legendary Pleistocene Ice Ages)
- nonstationary behavior, related to the well-known mid-Pleistocene transition from a “41k world” to a “100k world” somewhere around 0.8 Ma (?).

## 6 Conclusion

I wrote a strawman for this section. Please feel free to edit/modify and extend.

Version 1.0.0 of GeoChronR provides user-friendly access to common age-uncertain analyses in the paleogeosciences, along with intuitive visualization of the results. Although the focus has been on simplicity and ease-of-use, the GeoChronR also has the underlying infrastructure to support customized analyses for users seeking to address a more nuanced or complex question.



**Figure 5.** Need to write a caption

Nevertheless, in many ways the paleogeoscience community is only scratching the surface with age-uncertain analysis, and we look forward to working with the community to extend and expand the package. Looking forward, we suggest that the next major direction for age-uncertain analysis is philosophical, not technical. Thus far, the community (and GeoChronR) has focused on quantifying the range of possibilities presented by age-uncertainty, not on developing approaches to constrain which ensemble members are most likely. Theoretically, additional information from nearby records, forcings or covariance structures could do so, but little has been done to date (? , ).

GeoChronR is open-source community-software, and has benefitted substantially from multiple contributors and input from early adopters and workshop participants. We welcome feedback and strongly encourage contributions and enhancements.

Subsection text here.

### 6.0.1 Subsubsection Heading Here

Subsubsection text here.

## 7 Content section with citations

See the R Markdown docs for bibliographies and citations.

Copernicus supports biblatex. I put the .bib entries from the Paleocube proposal into `geochronr.bib`. Citations work like this:

Read (?), and (see ?).

## 8 Content section with R code chunks

You should always use `echo = FALSE` on R Markdown code blocks as they add formatting and styling not desired by Copernicus. The hidden workflow results in 42.

You can add verbatim code snippets without extra styles by using ````` without additional instructions.

```
sum <- 1 + 41
```

## 9 Content section with list

If you want to insert a list, you must

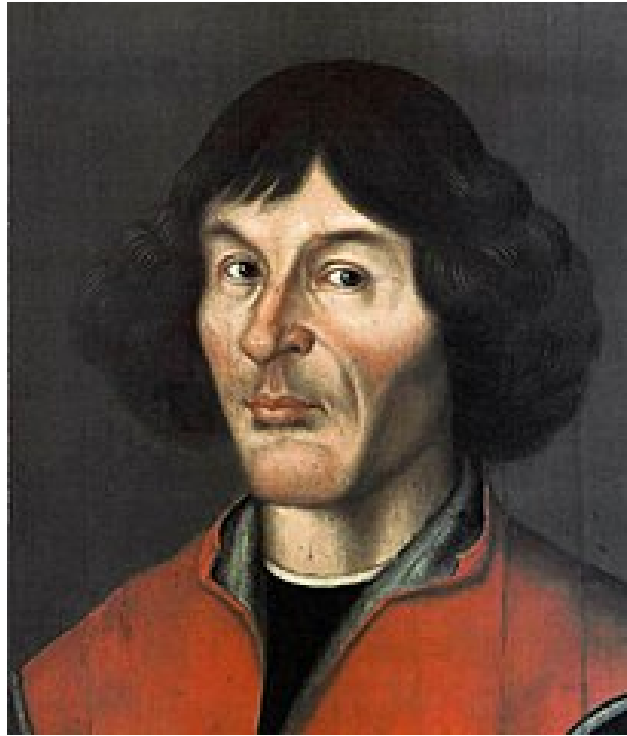
- leave
- empty lines
- between each list item
- leave
- empty lines
- between each list item

## 10 Examples from the official template

### 10.1 FIGURES

When figures and tables are placed at the end of the MS (article in one-column style), please add





**Figure 6.** one column figure

between bibliography and first table and/or figure as well as between each table and/or figure.

### **10.1.1 ONE-COLUMN FIGURES**

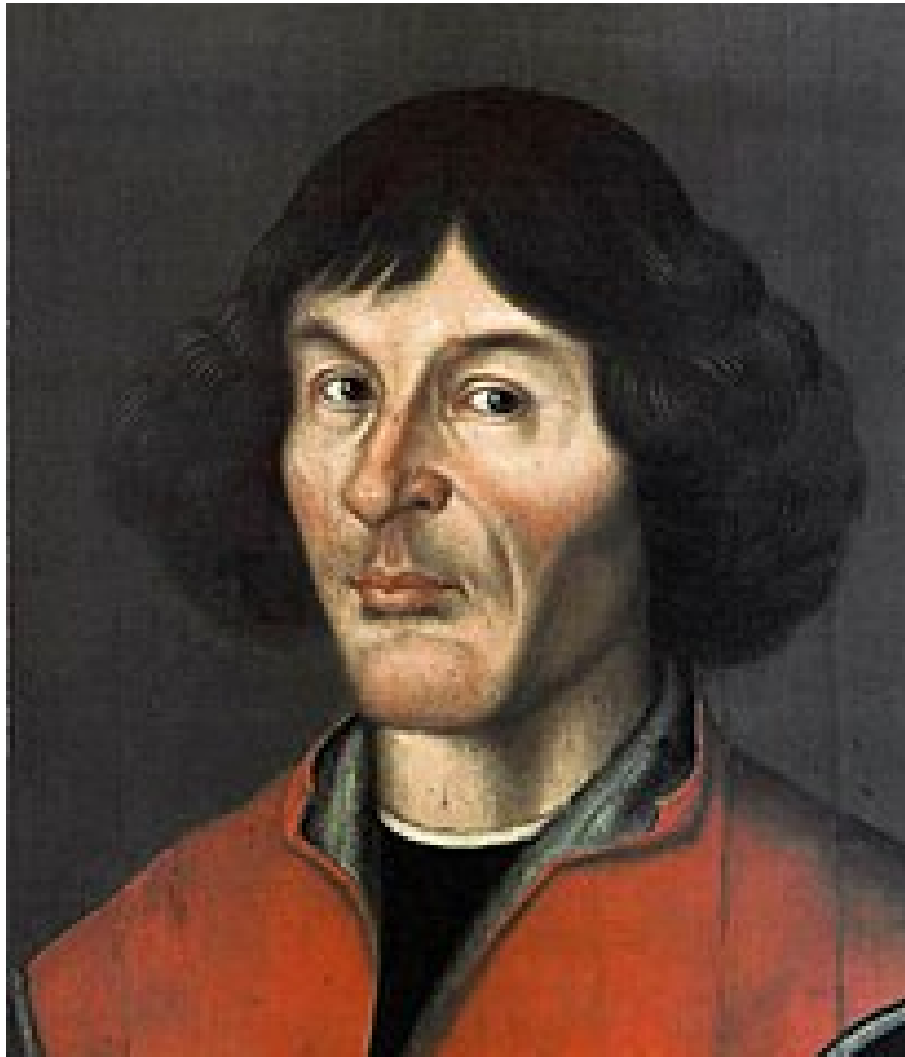
Include a 12cm width figure of Nikolaus Copernicus from Wikipedia with caption using R Markdown.

### **10.1.2 TWO-COLUMN FIGURES**

You can also include a larger figure.

## **10.2 TABLES**

You can add `\LaTeXtable` in an R Markdown document to meet the template requirements.



**Figure 7.** two column figure

**Table 1.** TEXT

a	b	c
1	2	3

Table Footnotes

**Table 2.** TEXT

a	b	c
1	2	3

Table footnotes

**10.2.1 ONE-COLUMN TABLE**

**10.2.2 TWO-COLUMN TABLE**

**10.3 MATHEMATICAL EXPRESSIONS**

All papers typeset by Copernicus Publications follow the math typesetting regulations given by the IUPAC Green Book (IUPAC: Quantities, Units and Symbols in Physical Chemistry, 2nd Edn., Blackwell Science, available at: <http://old.iupac.org/publications/book1993>).

- Physical quantities/variables are typeset in italic font (t for time, T for Temperature)
- Indices which are not defined are typeset in italic font (x, y, z, a, b, c)
- Items/objects which are defined are typeset in roman font (Car A, Car B)
- Descriptions/specifications which are defined by itself are typeset in roman font (abs, rel, ref, tot, net, ice)
- Abbreviations from 2 letters are typeset in roman font (RH, LAI)
- Vectors are identified in bold italic font using  $\boldsymbol{x}$
- Matrices are identified in bold roman font
- Multiplication signs are typeset using the LaTeX commands `\times` (for vector products, grids, and exponential notations) or `\cdot`
- The character `*` should not be applied as multiplication sign

## 10.4 EQUATIONS

### 10.4.1 Single-row equation

Unnumbered equations (i.e. using `$$` and getting inline preview in RStudio) are not supported by Copernicus.

$$1 \times 1 \cdot 1 = 42 \tag{1}$$

$$A = \pi r^2 \tag{2}$$

$$x = \frac{2b \pm \sqrt{b^2 - 4ac}}{2c}. \tag{3}$$

### 10.4.2 Multiline equation

$$3 + 5 = 8 \tag{4}$$

$$3 + 5 = 8 \tag{5}$$

$$3 + 5 = 8 \tag{6}$$

## 10.5 MATRICES

$$\begin{matrix} x & y & z \end{matrix}$$

$$\begin{matrix} x & y & z \end{matrix}$$

$$\begin{matrix} x & y & z \end{matrix}$$

## 10.6 ALGORITHM

If you want to use algorithms, you can either enable the required packages in the header (the default, see `algorithms: true`), or make sure yourself that the `LaTeX` packages `algorithms` and `algorithmicx` are installed so that `algorithm.sty` respectively `algorithmic.sty` can be loaded by the Copernicus template. Copernicus staff will remove all undesirable packages from your LaTeX source code, so please stick to using the header option, which only adds the two acceptable packages.

## 10.7 CHEMICAL FORMULAS AND REACTIONS

For formulas embedded in the text, please use `\chem{ }`, e.g.  $A \rightarrow B$ .

The reaction environment creates labels including the letter R, i.e. (R1), (R2), etc.

---

**Algorithm 1** Algorithm Caption

---

```
 $i \leftarrow 10$ 
if  $i \geq 5$  then
   $i \leftarrow i - 1$ 
else
  if  $i \leq 3$  then
     $i \leftarrow i + 2$ 
  end if
end if
```

---

- `\rightarrow` should be used for normal (one-way) chemical reactions
- `\rightleftharpoons` should be used for equilibria
- `\leftrightarrow` should be used for resonance structures



## 10.8 PHYSICAL UNITS

Please use `\unit{}` (allows to save the `math/$` environment) and apply the exponential notation, for example  $3.14\text{km h}^{-1}$  (using LaTeX mode: `\( 3.14\,, \unit{...} \)`) or  $0.872\text{ms}^{-1}$  (using only `\unit{0.872\,,m\,,s^{-1}}`).

## 11 Conclusions

The conclusion goes here. You can modify the section name with `\conclusions[modified heading if necessary]`.

*Code and data availability.* use this to add a statement when having data sets and software code available

## **Appendix A: Figures and tables in appendices**

### **A1 Option 1**

If you sorted all figures and tables into the sections of the text, please also sort the appendix figures and appendix tables into the respective appendix sections. They will be correctly named automatically.

### **A2 Option 2**

If you put all figures after the reference list, please insert appendix tables and figures after the normal tables and figures. Regarding figures and tables in appendices, the following two options are possible depending on your general handling of figures and tables in the manuscript environment: `\appendixfigures` needs to be added in front of appendix figures `\appendixtables` needs to be added in front of appendix tables To rename them correctly to A1, A2, etc., please add the following commands in front of them: Please add `\clearpage` between each table and/or figure. Further guidelines on figures and tables can be found below.

*Competing interests.* The authors declare no competing interests.

*Disclaimer.* disc

*Acknowledgements.* ack