

UThwgl - an R package for closed- and open-system uranium-thorium dating

Anthony Dosseto^{*,a}, Ben Marwick^b

^a*Wollongong Isotope Geochronology Laboratory, School of Earth, Atmosphere & Life Sciences, University of Wollongong, Wollongong NSW Australia*

^b*Department of Anthropology, University of Washington, Seattle, WA, USA*

Abstract

For several decades, uranium-thorium (U-Th) dating has allowed geochronologists to precisely date geological materials, providing invaluable geochronological constraints on Quaternary processes. Open-system dating of bones and teeth has also provided ages of human and faunal remains of archaeological significance.

To facilitate access of open-system U-Th dating to the broad scientific community, here we provide an R package, named *UThwgl*, that implements the Diffusion-Adsorption-Decay model of Sambridge et al. (2012). Description of input and output parameters is given, as well as a guide for running the model. The package can be used three different ways: (i) as a web application, (ii) through a web browser with an internet connection, or (iii) in R (most efficiently with RStudio). Examples of application of the model are also provided, showing that it yields ages within error of previously published values.

*Corresponding Author

Email addresses: tonyd@uow.edu.au (Anthony Dosseto), bmarwick@uw.edu (Ben Marwick)

1 Introduction

2 Open-system uranium-thorium (U-Th) dating of teeth and bones, while chal-
3 lenging, has revolutionised our ability to provide reliable chronology for humans
4 and fauna (Eggins et al. 2005; Grün et al. 2014; Sambridge, Grün, and Eggins
5 2012). Thus, this approach has significantly improved our understanding of hu-
6 man evolution (e.g. Dirks et al. 2017; Sutikna et al. 2016). Uranium-thorium
7 dating is based on the premise that a material takes up U but no Th, so all the
8 ^{230}Th in the sample comes from decay of ^{238}U . If detrital Th is included to the
9 sample, a correction must be included to account for the fraction of ^{230}Th which
10 is detrital and not derived from ^{238}U decay. Another requirement is that there
11 is no gain or loss of ^{230}Th , ^{234}U or ^{238}U after formation of the material. While it
12 is often the case for many geological samples such as corals or speleothems, this
13 requirement is rarely met for teeth and bone (although enamel can sometimes be
14 quite impervious to isotope gain or loss). Thus, for teeth and bone, U-Th dating
15 requires to take into account open system behaviour. The diffusion-adsorption-
16 decay (DAD) model developed by Sambridge et al. (2012) was instrumental to
17 implement successfully open-system U-Th dating. It allows for advective and
18 diffusive transport of uranium and thorium isotopes, while include synchronous
19 radioactive decay. The software implementation was written in Fortran and is
20 available as a Java GUI (<http://www.earth.org.au/codes/iDaD/>). In this arti-
21 cle, we propose a R package which implements the model of Sambridge et al.
22 (2012).

23 The motivation for providing this model as an R package is to increase
24 the transparency, reproducibility, and flexibility of the analytical workflow for
25 computing U-Th ages. Currently it is difficult to include the Java GUI in a
26 fully scripted data analysis so the method for computing the DAD model is not
27 highly transparent. This can obscure steps where key decisions are made that
28 are important for others to see to verify the reliability of the analysis. Enabling
29 a scripted workflow for computational analysis of geoscience data is important
30 for improving the reproducibility of results. Reproducibility refers the ability to
31 recreate the results or retest the hypotheses leading to a scientific claim, either
32 by rerunning the same code used by the original authors, or by writing new
33 code. High rates of irreproducibility of research results have been estimated
34 in several fields and disciplines (Medical Sciences 2015; Freedman, Cockburn,
35 and Simcoe 2015; Institute 2013; Ioannidis 2005; Collaboration and others 2015;
36 Camerer et al. 2018, 2016). Consequently, the transparency, openness, and
37 reproducibility of results and methods are receiving increased attention, and
38 the norms of research in many fields are changing (Nosek et al. 2015; Miguel et
39 al. 2014; Marwick 2016).

40 There is strong interest in open, transparent, and reusable research in the
41 geoscience community (Gil et al. 2016) and substantial progress toward open
42 data has been made in the geosciences with the widespread use of data services
43 of NASA, USGS, NOAA and community-built data portals such as OneGeol-
44 ogy, EarthChem, RRUFF, PANGAEA, PaleoBioDB, and others (Kattge, Díaz,
45 and Wirth 2014; Ma 2018). However, the use of open source software such

46 as R (Pebesma, Nüst, and Bivand 2012), and sharing of scripted data analy-
 47 sis workflows with research publications is not yet widespread (Hutton et al.
 48 2016). With this R package our goal is to make scripted and reproducible data
 49 analysis easy for open-system uranium-thorium dating. This will improve the
 50 transparency of geochronology research, and provide a more credible and robust
 51 foundation for scientific advancement (Hutton et al. 2016).

52 To enable re-use of our materials and improve reproducibility and trans-
 53 parency, all the results and visualisations in this paper can be reproduced using
 54 the RMarkdown vignette document included with the UThwgl package. We
 55 have archived these files at <http://doi.org/10.17605/OSF.IO/D5P7S> to ensure
 56 long-term accessibility. Our code is released under the MIT licence, our data as
 57 CC-0, and our figures as CC-BY, to enable maximum re-use (for more details,
 58 see Marwick 2016).

59 Methods

60 Data required for the DAD model are ($^{230}\text{Th}/^{238}\text{U}$) and ($^{234}\text{U}/^{238}\text{U}$) activity
 61 ratios collected along a transect perpendicular to the surface of the tooth or
 62 bone (brackets denote activity ratios throughout this article). Sampling for
 63 analysis can be done by micro-drilling or laser ablation. If the former, aliquots
 64 are then dissolved, followed by separation of U and Th using ion exchange
 65 chromatography. This is more time consuming (at least one week of work) than
 66 laser ablation, where the material sampled by the laser is directly sent to the
 67 mass spectrometer.

68 While laser ablation also offers a better spatial resolution than micro-drilling,
 69 the precision of the data is inferior because of the much smaller amount of
 70 material sampled. Uranium and thorium isotope ratios are then analysed by
 71 multi-collector inductively-coupled plasma mass spectrometry. A plasma ionise
 72 all U and Th atoms, their isotopes are separated through a magnetic field and
 73 each collected in a different collector. If using laser ablation, it is best to have
 74 two ion counters so ^{230}Th and ^{234}U can be collected simultaneously.

75 The distance of each analysis location from the inner and outer surfaces of
 76 the bones, for instance, needs to be recorded. One surface is given a coordinate
 77 of 1 and the other one -1, thus coordinates of analyses take values in between
 78 (Figure 1).



Figure 1: Modern human femur (132A/LB/27D/03) from Liang Bua, Flores, Indonesia. Two analysis transects can be seen. For a given transect, the outer and inner surface of the bone are given 1 and -1 reference coordinates, and the position of each analysis is calculated accordingly. Modified from Sutikna et al. (2016)

79 Working with the package

80 We provide three methods for using this package to suit different levels of
 81 familiarity with the R programming language. The simplest way to use the
 82 package is our web application (Figure 2). Using the web application requires
 83 no familiarity with R. To use the web application we upload a CSV file, then
 84 click through a series of tabs to inspect the data, adjust the model parameters,
 85 run the model, and inspect the output. The interface is mouse-driven and
 86 requires no programming. In the web application we upload the data file on the
 87 *Load the data* tab, set parameters from the *Set model parameters* tab, run the
 88 model by clicking the button *Run Simulation* on the same tab, and observe the
 89 results on the *Visualise the model* and *Inspect the model* tabs. We can change
 90 the parameters and re-run the model by click the button *Run Simulation*. Once
 91 done, close the window.

92 The second way to use the package is with Binder, a browser-based instance
 93 of R and RStudio that includes our package ready to work with. Binder is a
 94 server technology that turns computational material, such as an R package, into
 95 interactive computational environments in the cloud. Using Binder requires a
 96 novice level of familiarity with R, for example to use the code in this paper and
 97 adapt it to work with a different CSV file. Because Binder provides a complete
 98 R environment, custom R code can be written during a Binder instance to
 99 further explore the model's output in the browser. These two methods, the
 100 web application and Binder, do not require any software to be downloaded and

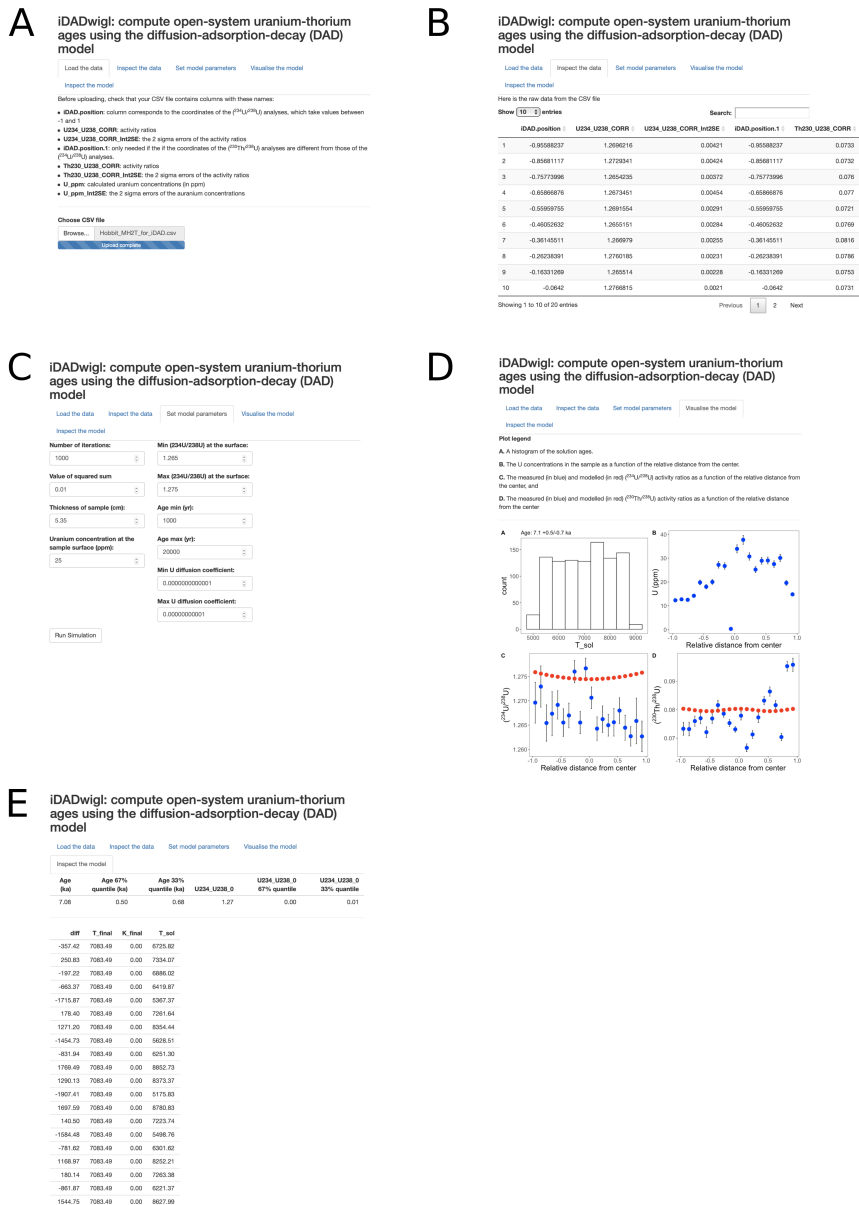


Figure 2: Screenshots of the web application for using the UThwgl package. A: Upload a CSV file of the data to model, B: Inspect a table of the uploaded data. C: Set the model parameters and run the model. D: Inspect visualisations of the model's output. E: Inspect and download the numeric output from the model.

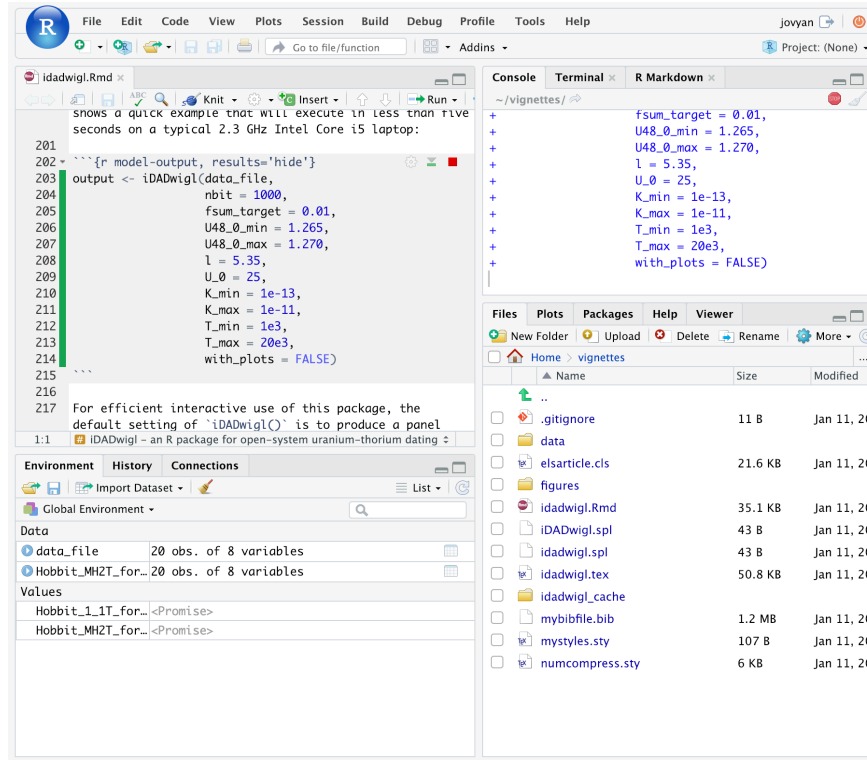


Figure 3: Screenshot of Binder running R and RStudio in a web browser window.

installed on the user’s computer, all computation occurs in the browser. The web application and Binder are suitable for getting a quick start on working with the package, but they require a connection to the internet, and they have limited memory and compute time available per instance.

The third method is to download and install the package locally to the user’s computer, and work with it in the user’s local installation of R and RStudio. This method requires some familiarity with R, but gives the most flexibility when working with the model because we are not limited by the memory and compute time of the cloud services. Our recommendation is to use Binder or a local installation of UThwgl because then the user can save an R script file that includes the name of the input file, the specific parameters used to generate the model output, and any downstream processing and visualisation. This script file and the CSV file can then be archived in a data repository to ensure long-term accessibility for other researchers.

Installing and attaching the package

First the user will need to download and install R, and we also recommend downloading and installing RStudio. To run the model, start RStudio and

118 install the package from GitHub. There are many ways to do this, one sim-
119 ple method is shown in the line below. This only needs to be done once per
120 computer.

```
source("https://install-github.me/tonydoss/UThwigl")
```

121 For routine data analysis, R scripts need to contain the following line to
122 attach the package to the current working environment. This line needs to be
123 run at the start of each analysis:

```
# attach the package  
library(UThwigl)
```

124 Alternatively, the model can be run from the user's browser without in-
125 stalling R or RStudio with this link.

126 Input data format

127 The key function of our package, `osUTh()` requires a data frame (a form of
128 table in R) with the following column names:

- 129 • `iDAD.position`
- 130 • `U234_U238_CORR`
- 131 • `U234_U238_CORR_Int2SE`
- 132 • `iDAD.position.1`
- 133 • `Th230_U238_CORR`
- 134 • `Th230_U238_CORR_Int2SE`
- 135 • `U_ppm`
- 136 • `U_ppm_Int2SE`

137 To help with preparing data for input into our function, we have included
138 two examples of input files. Inspecting the included data sets will be helpful for
139 understanding how to prepare new data to prepare for use with this package.
140 After attaching the package, we can access the built-in datasets with the `data()`
141 function, like this:

```
# access the data included in the UThwgl package  
data("Hobbit_1_1T_for_iDAD")  
data("Hobbit_MH2T_for_iDAD")
```

142 This will make the built-in data available in the R environment to inspect
143 and explore how to use the `osUTh()` function.

144 To use new data with this package, we need to import a CSV or Excel file
145 with the U-Th data into the R environment using a generic function such as
146 `read.csv` or `read_excel` from the `readxl` package (Wickham and Bryan 2018).
147 The code chunk below shows how to read one of the CSV files included in the
148 package into the R environment. This is a good method to use if the user
149 supplies their own CSV file to use with `osUTh()`. In this case we assume that
150 the user's working directory contains a directory called `data` and our CSV file
151 is in this `data` directory:

```
# read in one of the example CSV files included in the package  
data_file <-  
read.csv('data/Hobbit_MH2T_for_iDAD.csv')
```


Table 1 shows the data contained in the `Hobbit_MH2T_for_iDAD.csv` file included in the package

| iDAD.position | U234_U238_CORR | U234_U238_CORR_Int2SE | iDAD.position.1 | Th230_U238_CORR | Th230_U238_CORR_Int2SE | U_ppm | U_ppm_Int2SE |
|---------------|----------------|-----------------------|-----------------|-----------------|------------------------|-------|--------------|
| -0.956 | 1.270 | 0.004 | -0.956 | 0.073 | 0.002 | 12.3 | 0.6 |
| -0.857 | 1.273 | 0.004 | -0.857 | 0.073 | 0.002 | 12.7 | 0.6 |
| -0.758 | 1.265 | 0.004 | -0.758 | 0.076 | 0.002 | 12.5 | 0.6 |
| -0.659 | 1.267 | 0.005 | -0.659 | 0.077 | 0.002 | 14.2 | 0.7 |
| -0.560 | 1.269 | 0.003 | -0.560 | 0.072 | 0.002 | 19.8 | 1.0 |
| -0.461 | 1.266 | 0.003 | -0.461 | 0.077 | 0.002 | 18.0 | 0.9 |
| -0.361 | 1.267 | 0.003 | -0.361 | 0.082 | 0.002 | 20.0 | 1.0 |
| -0.262 | 1.276 | 0.002 | -0.262 | 0.079 | 0.001 | 27.2 | 1.4 |
| -0.163 | 1.266 | 0.002 | -0.163 | 0.075 | 0.001 | 26.7 | 1.3 |
| -0.064 | 1.277 | 0.002 | -0.064 | 0.073 | 0.001 | 0.3 | 0.0 |
| 0.035 | 1.271 | 0.002 | 0.035 | 0.078 | 0.002 | 33.9 | 1.7 |
| 0.134 | 1.264 | 0.002 | 0.134 | 0.067 | 0.001 | 37.7 | 1.9 |
| 0.233 | 1.266 | 0.003 | 0.233 | 0.071 | 0.001 | 30.7 | 1.5 |
| 0.332 | 1.265 | 0.002 | 0.332 | 0.077 | 0.002 | 25.2 | 1.3 |
| 0.431 | 1.266 | 0.003 | 0.431 | 0.083 | 0.001 | 28.9 | 1.4 |
| 0.530 | 1.268 | 0.003 | 0.530 | 0.086 | 0.002 | 29.0 | 1.4 |
| 0.629 | 1.264 | 0.003 | 0.629 | 0.082 | 0.002 | 27.5 | 1.4 |
| 0.728 | 1.263 | 0.002 | 0.728 | 0.070 | 0.001 | 30.1 | 1.5 |
| 0.827 | 1.266 | 0.005 | 0.827 | 0.095 | 0.002 | 19.6 | 1.0 |
| 0.926 | 1.263 | 0.003 | 0.926 | 0.096 | 0.002 | 14.8 | 0.7 |

Table 1: Data contained in the example CSV file `Hobbit_MH2T_for_iDAD.csv` included in the package

The columns `iDAD.position`, `U234_U238_CORR`, `U234_U238_CORR_Int2SE`, `Th230_U238_CORR` and `Th230_U238_CORR_Int2SE` must be present in the input data frame with these exact names for the model to function. The `osUTh()` function will check if the input data frame has these columns, and will stop with an error message if it does not find these columns. The `names()` function can be used to update column names of a data frame to ensure they match the names that the model function requires. The order of the columns in the data frame is not important.

162 The `iDAD.position` column corresponds to the coordinates of the ($^{234}\text{U}/^{238}\text{U}$)
 163 analyses, which as indicated above take values between -1 and 1 (Figure 1). The
 164 second `iDAD.position.1` column is used if the coordinates of the ($^{230}\text{Th}/^{238}\text{U}$)
 165 analyses are different from those of the ($^{234}\text{U}/^{238}\text{U}$) analyses.
 166 Columns `U234_U238_CORR` and `U234_U238_CORR_Int2SE` are the ($^{234}\text{U}/^{238}\text{U}$)
 167 activity ratios and their 2σ errors. Columns `Th230_U238_CORR` and `Th230_U238_CORR_Int2SE`
 168 are the ($^{230}\text{Th}/^{238}\text{U}$) activity ratios and their 2σ errors.
 169 Columns `U_ppm` and `U_ppm_Int2SE` are the calculated uranium concentra-
 170 tions (in ppm) and their 2σ errors. Uranium concentrations are not necessary
 171 for the model but needed to display the U concentration profile in a figure.

172 Details of the input parameters

173 Our key function, `osUTh()` has several arguments that need to be set before
 174 we can get meaningful results.

175 `nbit` is the number of iterations. For the first run, set to 1.
 176 `fsum_target` is the sum of the squared differences between the calculated
 177 and observed activity ratios. Give it a low value to start with (e.g. 0.01). If
 178 script takes too long, try a higher value for `fsum_target`.

179 `U48_0_min` and `U48_0_max` are the minimum and maximum values allowed
 180 for the ($^{234}\text{U}/^{238}\text{U}$) activity ratio at the surface of the sample. Since ($^{234}\text{U}/^{238}\text{U}$)
 181 does not vary greatly over the time period generally studied, the values mea-
 182 sured near the surface of the sample can be used as a guide. These values
 183 can be adjusted if the model fit to the data is not optimal. For `Hobbit_1-1T`
 184 they are taken to be 1.360 and 1.375, and for `Hobbit_MH2T`, 1.265 and 1.270,
 185 respectively.

186 `l` is the thickness of the sample in centimeters. For `Hobbit_1-1T` it is 3.5
 187 cm, for `Hobbit_MH2T` it is 5.35 cm

188 `U_0` is the uranium concentration at the surface in ppm. This value does
 189 not significantly affect the model results and values from analyses near either
 190 surface of the sample can be used as a guide. For `Hobbit_1-1T` it is taken to
 191 be 15 ppm; for `Hobbit_MH2T`, 25 ppm.

192 `K_min` and `K_max` are the minimum and maximum values allowed for the
 193 uranium diffusion coefficient (in cm^2/s). Values between 10^{-13} and 10^{-11} cm^2/s
 194 are generally appropriate.

195 `T_min` and `T_max` are the minimum and maximum values for the age of the
 196 specimen (yr). If there is no estimated knowledge of the sample age, the range
 197 of values can be 1,000 to 500,000 yr and adjusted later. For `Hobbit_1-1T`,
 198 in the final model run, they are taken to be 50,000 and 100,000 yr, and for
 199 `Hobbit_MH2T`, 1,000 and 20,000 yr, respectively.

200 After setting the U480 maximum and minimum values, run the function and
 201 adjust these min and max values by looking at the calculated `U48_0_final`,
 202 `K_final`, and `T_final`. Adjust `T_min` and `T_max` using first estimates of the
 203 age. As we iterate, increase the `nbit` value to reduce the error.

204 How to run the model

205 Attach the package as shown above and then run `osUTh()`, specifying the
 206 input data frame and the input parameters as described above. The code block
 207 below shows a quick example that will execute in less than five seconds on a
 208 typical 2.3 GHz Intel Core i5 laptop:

```
output <- osUTh(data_file,
  nbit = 1000,
  fsum_target = 0.01,
  U48_0_min = 1.265,
  U48_0_max = 1.270,
  l = 5.35,
  U_0 = 25,
  K_min = 1e-13,
  K_max = 1e-11,
  T_min = 1e3,
  T_max = 20e3,
  with_plots = FALSE)
```

209 For efficient interactive use of this package, the default setting of `osUTh()` is
 210 to produce a panel plot as seen in Figure 4(Figure 4. The setting `with_plots`
 211 `= FALSE` prevents plots from being generated which is more useful when the
 212 function is part of a longer sequence of code. The function runs faster when not
 213 producing pots, which is helpful when replicating many runs.

214 When run on the R console, this function will print a confirmation that the
 215 input data frame has the required columns, and print the resulting age value
 216 with an error reported as the 67% and 33% quantiles, for instance:

```
217 All required columns are present in the input data
218 [1] "Age: 7 +0.6/-0.7 ka"
```

219 The model computes a Monte Carlo simulation where age of the sample,
 220 U diffusion coefficient and ($^{234}\text{U}/^{238}\text{U}$) ratio at the surface of the sample are
 221 taken randomly within the range of values allowed. Results are only kept if the
 222 calculated sum of the squared differences between the calculated and observed
 223 activity ratios is less than the value set in `fsum_target`. If this is the case,
 224 the calculated ratios and the set of solutions for age of the sample, U diffusion
 225 coefficient and ($^{234}\text{U}/^{238}\text{U}$) ratio at the surface of the sample are saved. The
 226 model stops once the number of sets of solutions reaches `nbit`.

227 The final calculated age `T_final` (in yr), U diffusion coefficient `K_final` (in
 228 cm^2/s) and ($^{234}\text{U}/^{238}\text{U}$) ratio at the surface of the sample `U48_0_final` are the
 229 set of solutions where the solution age is the closest to the median age of the
 230 population of solutions. The uncertainty on each output paramter is calculated
 231 as the 67% and 33% quantiles of the population of solution sets.

232 In a typical analysis we will explore the model fit by first running the model
233 with a single iteration `nbit` and a small value for `fsum_target`, and then chang-
234 ing the range of allowed values for the ($^{234}\text{U}/^{238}\text{U}$) ratio at the surface and the
235 age of the sample. Once we obtain a satisfying fit (by visual inspection of the
236 produced figures), we would increase `nbit` to a higher value (e.g. 1000) and run
237 the model one last time. See below for an example of analysis.

238 *Inspecting the model's output*

239 `T_final`, `K_final` and `U48_0_final` are included in the model's output,
 240 along with their uncertainties. The function also includes a one-row data frame
 241 summarising the age:

| Age (ka) | Age 67% quantile (ka) | Age 33% quantile (ka) | U234_ U238_0 | U234_ U238_0 67% quantile | U234_ U238_0 33% quantile |
|----------|-----------------------|-----------------------|--------------|---------------------------|---------------------------|
| 7.02 | 0.59 | 0.58 | 1.2651 | 0.0049 | 0.0001 |

Table 2: Summary table of the computed age and error values

242 The last item in the output is a copy of the input data with two additional
 243 columns, the calculated activity ratios, ($^{234}\text{U}/^{238}\text{U}$) and ($^{230}\text{Th}/^{238}\text{U}$), for each
 244 measurement location on the sample.

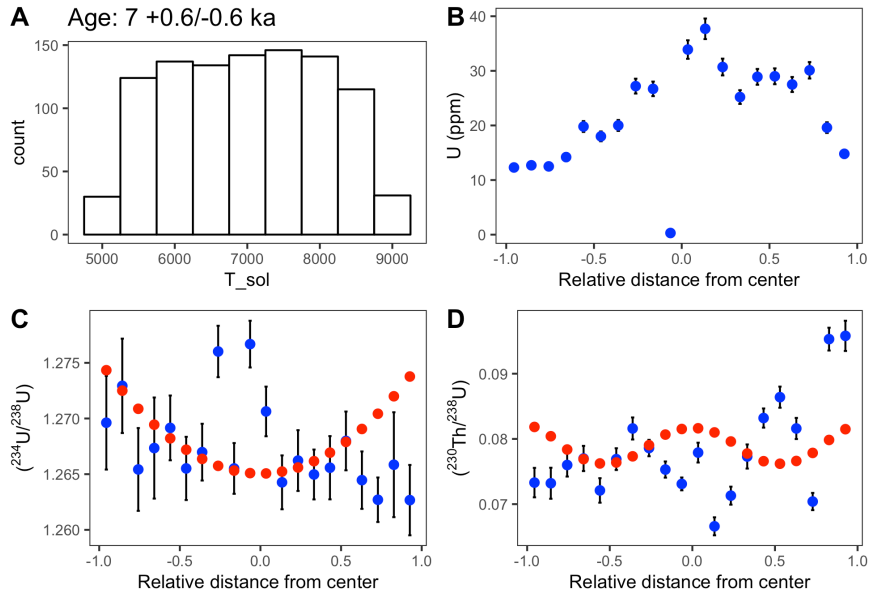


Figure 4: Example of the visualisations produced by the `osUTh()` function, using the demonstration run described above. A: Histogram of the solution ages, B: Uranium concentration profile for transect 2 of modern human femur 132A/LB/27D/03. C: Calculated (red) and observed (blue) $(^{234}\text{U}/^{238}\text{U})$ activity ratios for transect 2 of modern human femur 132A/LB/27D/03. D: Calculated (red) and observed (blue) $(^{230}\text{Th}/^{238}\text{U})$ activity ratios for transect 2 of modern human femur 132A/LB/27D/03.

Visualising the model's output

`osUTh()` returns several figures useful for visualisation of the model results along with the data:

1. a histogram of the solution ages (Figure 4 A)
2. the U concentrations in the sample as a function of the relative distance from the center (Figure 4 B)
3. the measured (in blue) and modelled (in red) $(^{234}\text{U}/^{238}\text{U})$ activity ratios as a function of the relative distance from the center (Figure 4 C), and
4. the measured (in blue) and modelled (in red) $(^{230}\text{Th}/^{238}\text{U})$ activity ratios as a function of the relative distance from the center (Figure 4 D).

| iDAD.position | U234_U238_CORR | U234_U238_CORR_Int2SE | iDAD.position.1 | Th230_U238_CORR | Th230_U238_CORR_Int2SE | U_ppm | U_ppm_Int2SE | U234_U238_CALC | Th230_U238_CALC |
|---------------|----------------|-----------------------|-----------------|-----------------|------------------------|-------|--------------|----------------|-----------------|
| -0.956 | 1.270 | 0.004 | -0.956 | 0.073 | 0.002 | 12.3 | 0.6 | 1.274 | 0.082 |
| -0.857 | 1.273 | 0.004 | -0.857 | 0.073 | 0.002 | 12.7 | 0.6 | 1.273 | 0.080 |
| -0.758 | 1.265 | 0.004 | -0.758 | 0.076 | 0.002 | 12.5 | 0.6 | 1.271 | 0.078 |
| -0.659 | 1.267 | 0.005 | -0.659 | 0.077 | 0.002 | 14.2 | 0.7 | 1.269 | 0.077 |
| -0.560 | 1.269 | 0.003 | -0.560 | 0.072 | 0.002 | 19.8 | 1.0 | 1.268 | 0.076 |
| -0.461 | 1.266 | 0.003 | -0.461 | 0.077 | 0.002 | 18.0 | 0.9 | 1.267 | 0.076 |
| -0.361 | 1.267 | 0.003 | -0.361 | 0.082 | 0.002 | 20.0 | 1.0 | 1.266 | 0.077 |
| -0.262 | 1.276 | 0.002 | -0.262 | 0.079 | 0.001 | 27.2 | 1.4 | 1.266 | 0.079 |
| -0.163 | 1.266 | 0.002 | -0.163 | 0.075 | 0.001 | 26.7 | 1.3 | 1.265 | 0.081 |
| -0.064 | 1.277 | 0.002 | -0.064 | 0.073 | 0.001 | 0.3 | 0.0 | 1.265 | 0.082 |
| 0.035 | 1.271 | 0.002 | 0.035 | 0.078 | 0.002 | 33.9 | 1.7 | 1.265 | 0.082 |
| 0.134 | 1.264 | 0.002 | 0.134 | 0.067 | 0.001 | 37.7 | 1.9 | 1.265 | 0.081 |
| 0.233 | 1.266 | 0.003 | 0.233 | 0.071 | 0.001 | 30.7 | 1.5 | 1.266 | 0.080 |
| 0.332 | 1.265 | 0.002 | 0.332 | 0.077 | 0.002 | 25.2 | 1.3 | 1.266 | 0.078 |
| 0.431 | 1.266 | 0.003 | 0.431 | 0.083 | 0.001 | 28.9 | 1.4 | 1.267 | 0.077 |
| 0.530 | 1.268 | 0.003 | 0.530 | 0.086 | 0.002 | 29.0 | 1.4 | 1.268 | 0.076 |
| 0.629 | 1.264 | 0.003 | 0.629 | 0.082 | 0.002 | 27.5 | 1.4 | 1.269 | 0.077 |
| 0.728 | 1.263 | 0.002 | 0.728 | 0.070 | 0.001 | 30.1 | 1.5 | 1.270 | 0.078 |
| 0.827 | 1.266 | 0.005 | 0.827 | 0.095 | 0.002 | 19.6 | 1.0 | 1.272 | 0.080 |
| 0.926 | 1.263 | 0.003 | 0.926 | 0.096 | 0.002 | 14.8 | 0.7 | 1.274 | 0.081 |

Table 3: Example of output table including the input data described above, and two new columns

255 **Case study of two ages from Sutikna et al. 2016**

256 The package includes two sample data sets derived from Sutikna et al.
 257 (2016) : “Hobbit_MH2T_for_iDAD.csv” is data from transect 2 for mod-
 258 ern human femur 132A/LB/27D/03 (shown above in Table 1). “Hobbit_1-
 259 1T_for_iDAD.csv” is data from transect 1 for *Homo floresiensis* ulna LB1/52
 260 (Table 4). For the latter, six analyses were removed from the set as in Sutikna
 261 et al. (2016).

| iDAD_position | U234_U238_CORR | U234_U238_CORR_Int2SE | iDAD_position.1 | Th230_U238_CORR | Th230_U238_CORR_Int2SE | U_ppm | U_ppm_Int2SE |
|---------------|----------------|-----------------------|-----------------|-----------------|------------------------|-------|--------------|
| -0.143 | 1.369 | 0.002 | -0.143 | 0.699 | 0.006 | 32.0 | 1.6 |
| 0.165 | 1.370 | 0.002 | 0.165 | 0.733 | 0.008 | 41.1 | 2.1 |
| 0.319 | 1.364 | 0.002 | 0.319 | 0.672 | 0.006 | 35.8 | 1.8 |
| 0.472 | 1.362 | 0.003 | 0.472 | 0.636 | 0.006 | 27.6 | 1.4 |
| 0.626 | 1.365 | 0.003 | 0.626 | 0.641 | 0.006 | 31.0 | 1.6 |
| 0.780 | 1.374 | 0.003 | 0.780 | 0.712 | 0.005 | 27.9 | 1.4 |

Table 4: Data contained in the example CSV file Hobbit_11T_for_iDAD.csv included in the package

262 *Age of the modern human remains from Sutikna et al. 2016*

263 For transect 2 of 132A/LB/27D/03, Sutikna et al. (2016) reported an age
 264 of 7.4 ± 0.5 ka (thousand years before 2014). With UThwgl, we first run
 265 the model with `nbit = 1`, `fsum_target = 0.05`, `U48_0_min` and `U48_0_max`
 266 `= 1.25` and `1.3`, respectively, `l = 5.35` cm, `U_0 = 25` ppm, `K_min` and `K_max`
 267 `= 10^{-13}` and `10^{-11}` cm²/s, respectively, `T_min` and `T_max = 10^3` and `500×10^3`
 268 yr, respectively. `U48_0_min` and `U48_0_max` are determined by considering the
 269 measured (²³⁴U/²³⁸U) values near the surfaces of the sample. `T_min` and `T_max`
 270 values were chosen such that no a priori knowledge of the age biases the results.

271 With this first run, we obtain an age of 10 ka. There is no calculated error
 272 on the age since there is only one iteration. The calculated (²³⁴U/²³⁸U) and
 273 (²³⁰Th/²³⁸U) ratios can be quite different from observed values (Figure 6). For
 274 the (²³⁴U/²³⁸U), it makes sense to thus use a narrower range for `U48_0_min` and
 275 `U48_0_max`. Higher calculated (²³⁰Th/²³⁸U) ratios compared to observed values
 276 suggests that the calculated age is too old (since this ratio increases with age),

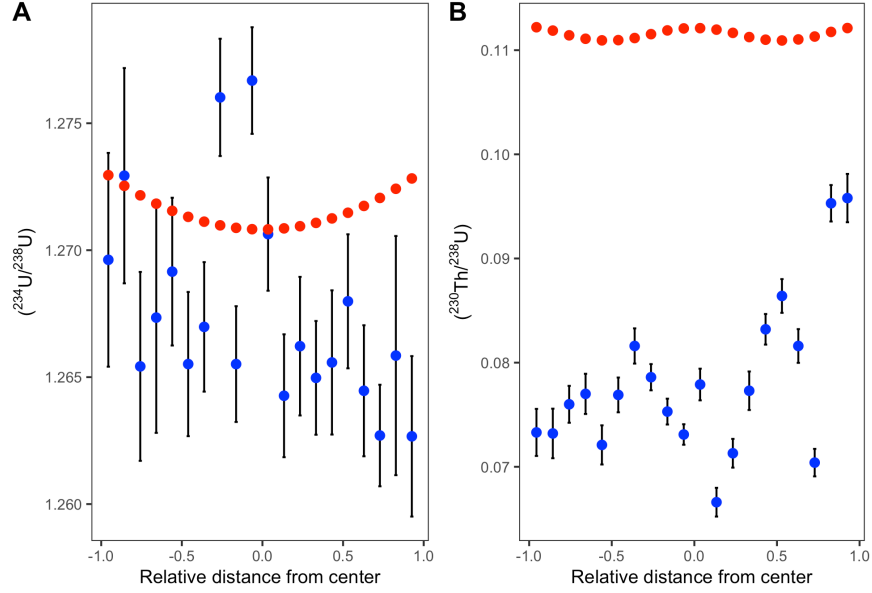


Figure 5: Results from the model's first run with the modern human femur. A: Calculated (red) and observed (blue) $(^{234}\text{U}/^{238}\text{U})$ activity ratios for transect 2 of modern human femur 132A/LB/27D/03. B: Calculated (red) and observed (blue) $(^{230}\text{Th}/^{238}\text{U})$ activity ratios for transect 2 of modern human femur 132A/LB/27D/03.

277 and the opposite if the calculated $(^{230}\text{Th}/^{238}\text{U})$ ratios are too low. Thus, we
 278 should adjust `Tmix` and/or `Tmax` accordingly.

279 Since the first run suggests a Holocene age for the sample, the measured
 280 $(^{234}\text{U}/^{238}\text{U})$ at the surfaces must be similar to the calculated values, thus the
 281 chosen values for the range above. Once `U48_0_min`, `U48_0_max`, `Tmin` and
 282 `Tmax` parameters have been adjusted, the model is run again. `fsum_target`
 283 can also be decreased to 0.01 in order to get a better fit and error, but it is at
 284 the expense of computing time. This operation is repeated until a satisfying
 285 fit is obtained (by visual inspection of the figures). Finally, the model is run
 286 once more, increasing the number of iterations to 1000 (or more). Following
 287 this method, we obtain an age of $6.3 \pm 1.2/-1.1$ ka (Figure 6).

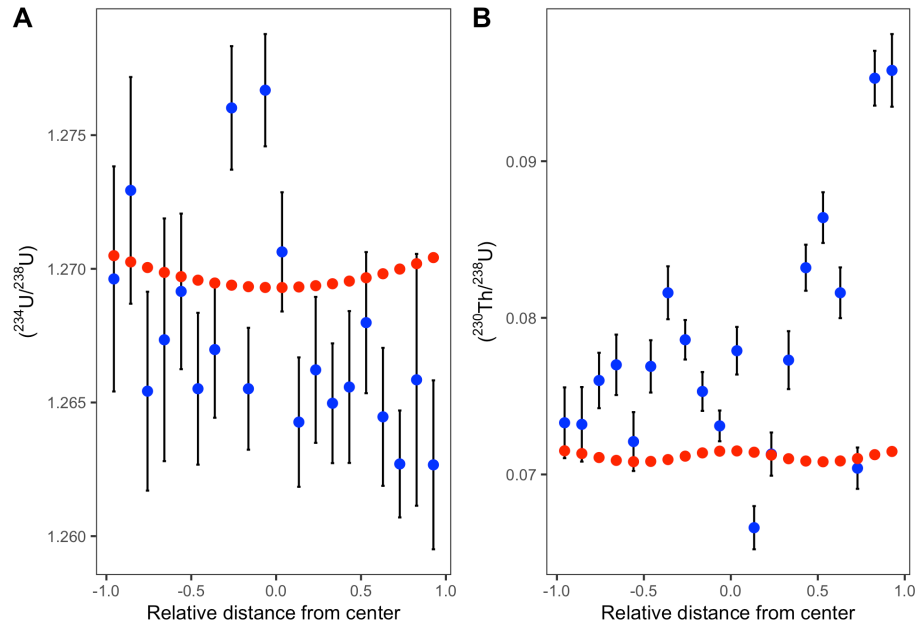


Figure 6: Results from the model's second run with the modern human femur. A: Calculated (red) and observed (blue) $(^{234}\text{U}/^{238}\text{U})$ activity ratios for transect 2 of modern human femur 132A/LB/27D/03. B: Calculated (red) and observed (blue) $(^{230}\text{Th}/^{238}\text{U})$ activity ratios for transect 2 of modern human femur 132A/LB/27D/03.

288 *Age of the Homo floresiensis remains from Sutikna et al. 2016*

289 For transect 1 of LB1/52, Sutikna et al. (2016) reported an age of 79.0 ± 3.7
290 ka. With osUth, using data in the file `Hobbit_1-1T_for_iDAD.csv` provided in
291 the package, and following the same method as above, we obtain an age of 75.4
292 $\pm 0.9/-1$ ka (Figure 7).

293 Note that results and errors will vary slightly for each run since populations
294 are solution sets are randomly generated.

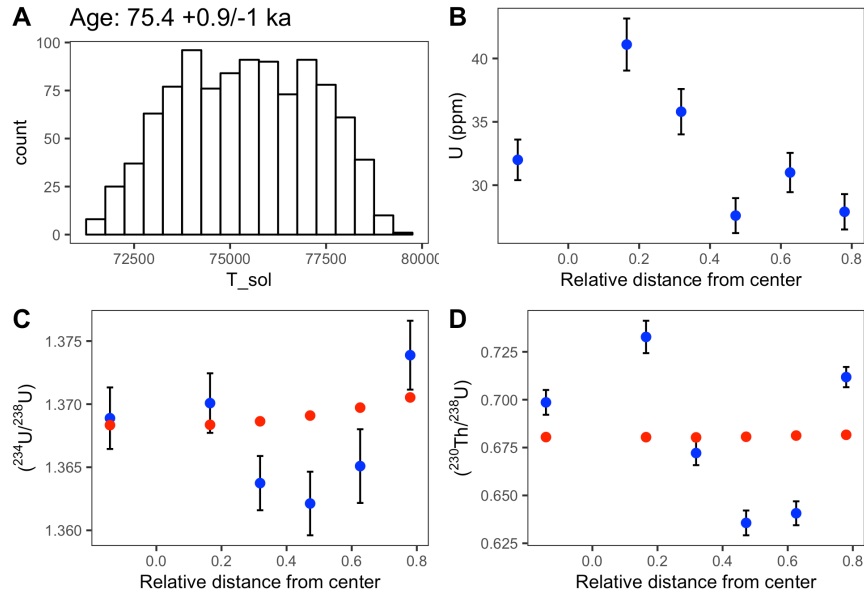


Figure 7: Results from running the model with *Homo floresiensis* ulna LB1/52 data from Sutikna et al. (2016). A: Histogram of the solution ages, B: Uranium concentration profile for transect 1 of *Homo floresiensis* ulna LB1/52. C: Calculated (red) and observed (blue) $(^{234}\text{U}/^{238}\text{U})$ activity ratios for transect 1 of *Homo floresiensis* ulna LB1/52. D: Calculated (red) and observed (blue) $(^{230}\text{Th}/^{238}\text{U})$ activity ratios for transect 1 of *Homo floresiensis* ulna LB1/52.

Conclusions

In this paper we have described `UThwig1`, an open source R package for computation of open-system U-Th ages. This helps to enable transparency, reproducibility, and flexibility of the analytical workflow for computing U-Th ages. The examples above show that results from our model are within error of previously published ages.

References

- Camerer, Colin F., Anna Dreber, Eskil Forsell, Teck-Hua Ho, Jürgen Huber, Magnus Johannesson, Michael Kirchler, et al. 2016. “Evaluating Replicability of Laboratory Experiments in Economics.” *Science* 351 (6280): 1433–6. <https://doi.org/10.1126/science.aaf0918>.
- Camerer, Colin F., Anna Dreber, Felix Holzmeister, Teck-Hua Ho, Jürgen Huber, Magnus Johannesson, Michael Kirchler, et al. 2018. “Evaluating the Replicability of Social Science Experiments in Nature and Science Between 2010 and 2015.” *Nature Human Behaviour* 2 (9): 637.
- Collaboration, Open Science, and others. 2015. “Estimating the Reproducibility of Psychological Science.” *Science* 349 (6251): aac4716.
- Dirks, P. H., E. M. Roberts, H. Hilbert-Wolf, J. D. Kramers, J. Hawks, A. Dosseto, M. Duval, et al. 2017. “The Age of Homo Naledi and Associated Sediments in the Rising Star Cave, South Africa.” Journal Article. *Elife* 6. <https://doi.org/10.7554/eLife.24231>.
- Eggins, Stephen M., Rainer Grün, Malcolm T. McCulloch, Alistair W. G. Pike, John Chappell, Les Kinsley, Graham Mortimer, et al. 2005. “In Situ U-Series Dating by Laser-Ablation Multi-Collector Icpms: New Prospects for Quaternary Geochronology.” Journal Article. *Quaternary Science Reviews* 24 (23-24): 2523–38. <https://doi.org/10.1016/j.quascirev.2005.07.006>.
- Freedman, Leonard P, Iain M Cockburn, and Timothy S Simcoe. 2015. “The Economics of Reproducibility in Preclinical Research.” *PLoS Biology* 13 (6): e1002165.
- Gil, Yolanda, Cédric H. David, Ibrahim Demir, Bakinam T. Essawy, Robinson W. Fulweiler, Jonathan L. Goodall, Leif Karlstrom, et al. 2016. “Toward the Geoscience Paper of the Future: Best Practices for Documenting and Sharing Research from Data to Software to Provenance.” *Earth and Space Science* 3 (10): 388–415. <https://doi.org/10.1002/2015EA000136>.
- Grün, Rainer, Stephen Eggins, Leslie Kinsley, Hannah Moseley, and Malcolm Sambridge. 2014. “Laser Ablation U-Series Analysis of Fossil Bones and Teeth.” Journal Article. *Palaeogeography, Palaeoclimatology, Palaeoecology* 416: 150–67. <https://doi.org/http://dx.doi.org/10.1016/j.palaeo.2014.07.023>.
- Hutton, Christopher, Thorsten Wagener, Jim Freer, Dawei Han, Chris Duffy, and Berit Arheimer. 2016. “Most Computational Hydrology Is Not Reproducible, so Is It Really Science?” *Water Resources Research* 52 (10): 7548–55. <https://doi.org/10.1002/2016WR019285>.
- Institute, Global Biological Standards. 2013. “The Case for Standards in Life Science Research: Seizing Opportunities at a Time of Critical Need.”
- Ioannidis, John PA. 2005. “Why Most Published Research Findings Are False.” *PLoS Medicine* 2 (8): e124.
- Kattge, Jens, Sandra Díaz, and Christian Wirth. 2014. “Of Carrots and Sticks.” *Nature Geoscience* 7 (October): 778.
- Ma, Xiaogang. 2018. “Data Science for Geoscience: Leveraging Mathematical Geosciences with Semantics and Open Data.” In *Handbook of Mathematical Geosciences*, 687–702. Springer.

- Marwick, Ben. 2016. “Computational Reproducibility in Archaeological Research: Basic Principles and a Case Study of Their Implementation.” *Journal of Archaeological Method and Theory*, 1–27. <https://doi.org/10.1007/s10816-015-9272-9>.
- Medical Sciences, Academy of. 2015. “Reproducibility and Reliability of Biomedical Research: Improving Research Practice.” In *Symposium Report*.
- Miguel, Edward, Colin Camerer, Katherine Casey, Joshua Cohen, Kevin M Esterling, Alan Gerber, Rachel Glennerster, et al. 2014. “Promoting Transparency in Social Science Research.” *Science* 343 (6166): 30–31.
- Nosek, Brian A, George Alter, George C Banks, Denny Borsboom, Sara D Bowman, Steven J Breckler, Stuart Buck, et al. 2015. “Promoting an Open Research Culture.” *Science* 348 (6242): 1422–5.
- Pebesma, Edzer, Daniel Nüst, and Roger Bivand. 2012. “The R Software Environment in Reproducible Geoscientific Research.” *Eos, Transactions American Geophysical Union* 93 (16): 163–63. <https://doi.org/10.1029/2012EO160003>.
- Sambridge, M., R. Grün, and S. Eggins. 2012. “U-Series Dating of Bone in an Open System: The Diffusion-Adsorption-Decay Model.” Journal Article. *Quaternary Geochronology*. <http://www.scopus.com/inward/record.url?eid=2-s2.0-84857958926&partnerID=40&md5=01e7f1b7d3554da26b247125425ff496>.
- Sutikna, Thomas, Matthew W. Tocheri, Michael J. Morwood, E. Wahyu Saptomo, Jatmiko, Rokus Due Awe, Sri Wasisto, et al. 2016. “Revised Stratigraphy and Chronology for Homo Floresiensis at Liang Bua in Indonesia.” Journal Article. *Nature* 532 (7599): 366–69. <https://doi.org/10.1038/nature17179>.
- Wickham, Hadley, and Jennifer Bryan. 2018. *Readxl: Read Excel Files*. <https://CRAN.R-project.org/package=readxl>.

Colophon

This report was generated on 2019-01-12 23:14:19 using the following computational environment and dependencies:

```
# which R packages and versions?
devtools::session_info()
```

```
## - Session info -----
## setting value
## version R version 3.5.1 (2018-07-02)
## os      macOS 10.14.2
## system  x86_64, darwin15.6.0
## ui      X11
## language (EN)
## collate C
## ctype   en_US.UTF-8
## tz      America/Los_Angeles
## date    2019-01-12
##
## - Packages -----
## ! package * version      date      lib
## assertthat 0.2.0      2017-04-11 [1]
## backports   1.1.3      2018-12-14 [1]
## bookdown    0.9        2018-12-21 [1]
## callr       3.1.1      2018-12-21 [1]
## cli         1.0.1      2018-09-25 [1]
## colorspace  1.3-2      2016-12-14 [1]
## commonmark  1.7        2018-12-01 [1]
## cowplot     * 0.9.4      2019-01-08 [1]
## crayon      1.3.4      2018-12-18 [1]
## desc        1.2.0      2018-05-01 [1]
## deSolve     1.21       2018-05-09 [1]
## devtools    2.0.1      2018-10-26 [1]
## digest      0.6.18     2018-10-10 [1]
## dplyr       * 0.7.99.9000 2018-12-25 [1]
## evaluate    0.12       2018-10-09 [1]
## fs          1.2.6      2018-08-23 [1]
## ggplot2     * 3.1.0.9000 2018-12-25 [1]
## glue        1.3.0      2018-10-31 [1]
## gtable      0.2.0      2016-02-26 [1]
## htmltools   0.3.6      2017-04-28 [1]
## knitr       * 1.21       2018-12-10 [1]
## labeling    0.3        2014-08-23 [1]
## lazyeval    0.2.1      2017-10-29 [1]
## magrittr    1.5        2014-11-22 [1]
## memoise     1.1.0      2017-04-21 [1]
```

```

##      munsell      0.5.0      2018-06-12 [1]
##      pillar      1.3.1.9000 2018-12-23 [1]
##      pkgbuild     1.0.2      2018-10-16 [1]
##      pkgconfig    2.0.2      2018-08-16 [1]
##      pkgload      1.0.2      2018-10-29 [1]
##      prettyunits  1.0.2      2015-07-13 [1]
##      processx     3.2.1      2018-12-05 [1]
##      ps           1.3.0      2018-12-21 [1]
##      purrr        0.2.5      2018-05-29 [1]
##      R6            2.3.0      2018-10-04 [1]
##      Rcpp         1.0.0      2018-11-07 [1]
##      remotes      2.0.2      2018-10-30 [1]
##      rlang        0.3.1      2019-01-08 [1]
##      rmarkdown    1.11      2018-12-08 [1]
##      roxygen2      6.1.1      2018-11-07 [1]
##      rprojroot    1.3-2      2018-01-03 [1]
##      rstudioapi    0.9.0      2019-01-09 [1]
##      rtticles      0.6       2018-10-09 [1]
##      scales       1.0.0      2018-08-09 [1]
##      sessioninfo   1.1.1      2018-11-05 [1]
##      stringi      1.2.4      2018-07-20 [1]
##      stringr      1.3.1      2018-05-10 [1]
##      testthat     2.0.1      2018-10-13 [1]
##      tibble       2.0.0      2019-01-04 [1]
##      tidyselect    0.2.5      2018-10-11 [1]
##      usethis       1.4.0      2018-08-14 [1]
## P UThwigl * 0.1.0      2019-01-13 [?]
##      withr        2.1.2      2018-03-15 [1]
##      xfun         0.4       2018-10-23 [1]
##      xml2         1.2.0.9000 2018-12-02 [1]
##      xtable       * 1.8-3      2018-08-29 [1]
##      yaml         2.2.0      2018-07-25 [1]
## source
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.2)
## Github (gaborcsardi/crayon@74bee76)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.1)
## CRAN (R 3.5.0)

```



```

## Github (tidyverse/dplyr@bee3351)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## Github (tidyverse/ggplot2@7f13dfa)
## Github (tidyverse/glue@35c61e9)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.1)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## Github (r-lib/pillar@9cc3030)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.2)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.1)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## CRAN (R 3.5.2)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## local
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
## Github (hadley/xml2@de9781d)
## CRAN (R 3.5.0)
## CRAN (R 3.5.0)
##
## [1] /Library/Frameworks/R.framework/Versions/3.5/Resources/library

```

```
##  
## P -- Loaded and on-disk path mismatch.
```

The current Git commit details are:

```
git2r::repository(here::here())
```

```
## Local:   master /Users/bmarwick/Desktop/UThwigl  
## Remote:  master @ origin (https://github.com/benmarwick/iDADwigl.git)  
## Head:    [d867ced] 2019-01-12: update website for pkg
```