AIMS Geosciences

2022, Volume 8, Issue 1: 16-32. doi: 10.3934/geosci.2022002

MatCalib: a Matlab software package for Bayesian modeling of radiocarbon ages subject to temporal order constraints

Shiyong Yu*

School of Geography, Geomatics, and Planning, Jiangsu Normal University, 101 Shanghai Road, Xuzhou, Jiangsu 221116, China

Correspondence: Email: syu@jsnu.edu.cn

User's Manual

Running the software package is quite straightforward. A tab delimited spread sheet template is provided, which allow users to enter their data, including laboratory codes, depths, ^{14}C ages and errors, reservoir ages and errors if any, specify the calibration curve (e.g., IntCal13 or IntCal20) for each age, and define the temporal ordering of the ages (e.g., "ordered", "unordered", or "coeval") in each period. The only file users need to modify is *MatCalib_main.m,* which provides a textual interface enabling users to read their data, define their project-specific problem (e.g., boundary relationships), specify parameters for MCMC simulations, and run the model to obtain, analyze, and visualize the results. In the Matlab environment, users need to open the main program *MatCalib_main*, give the nearest years that the modeled calendar ages are to be rounded to, choose the time scale in which the modeled calendar ages are to be reported, and specify the parameters for Monte Carlo simulations such as the number of chains to be run for convergence diagnoses as well as the length, burn-in period, and thinning interval of the chains. Then, MCMC simulations are conducted using the function *age_modeling* and the convergence of the chains is tested using the function *convergence.*

Upon completion of the MCMC simulations, the empirical probability density function, as well as the

68.2% and 95.4% HPD regions of the modeled calendar ages of the ^{14}C ages and the early and late boundaries of each period, are estimated using the method of Lougheed and Obrochta, and the results are saved automatically to files for subsequent analyses. Users can plot the results either against sample numbers using the function *plot_ages*, or against depths using the function *plot_age_depth* if depth information is provided. Users also can compare any two ages by calculating their posterior difference and calculate the pooled mean of several ages using the functions *age_difference* and *pooled_mean*, respectively. The results are also given in terms of the empirical probability density function as well as the 68.2% and 95.4% HPD regions, and they can be plotted using the functions *plot_difference* and *plot_pooled_mean*, respectively.

It is noteworthy that the MCMC method is simulation-based, which requires a large number of iterations to produce reliable (i.e., converged and well mixed) results. Therefore, the model may yield slightly different results at each time of a run. Another caveat is that models may be so inconsistent with the data that no possible solutions will be yielded, particularly in the case of the presence of extreme outliers. Therefore, users should choose and remove the outliers before running the model to produce valid results. In other cases, the solution space may be very fragmentary, providing that the MCMC solutions will be different with each different starting position. Keeping these properties of the MCMC method in mind, users must check the reproducibility by making multiple runs of the model with a different number of iterations, burn-in period, thinning size, and initial value (automatically reset for each run using the function *initialization*).