New features in the dendroTools R package: bootstrapped and partial correlation coefficients for monthly and daily climate data

Jernej Jevšenak[1, *]

[1]Slovenian Forestry Institute, Department of Forest Yield and Silviculture, Večna pot 2, 1000, Ljubljana, Slovenia

[*]correspondence: jernej.jevsenak@gozdis.si

Abstract: Climate-growth relationships are usually analysed using monthly climate data. The dendroTools R package also provides methodological approaches that enable climate-growth analysis for daily climate data. Such analysis reveals more complete climate signal patterns. In this article, new functions of the dendroTools R package are presented. Partial correlation coefficients are now implemented and can be used to calculate the strength of a linear relationship between two variables, while controlling for a third variable. Bootstrapped correlations can then be used to provide insights into the confidence intervals of statistical estimates. The calculation of partial and bootstrapped correlations is available for daily and monthly data. Finally, data transformation, S3 generic plotting and summary functions are also presented here.

Key words: dendroTools, daily climate data, partial correlations, bootstrap, dendroclimatology

1. Introduction

The R package dendroTools provides functions that enable dendroclimatological analysis using climate data on a daily scale. While alternative software such as CLIMTREG (Beck et al., 2013) and DendroCorr (Hulist et al., 2016) is available, the advantages of dendroTools are its implementation in the very popular R environment (R Core Team, 2019) and open source R code, which can also be modified to meet user specific needs. Using climate data on daily scales provides more flexible analysis of climate-growth relationships, such as climate reconstructions of periods not bounded by months and changes in climate signal patterns over time. Jevšenak (2019) compared climate-proxy correlations on a European-wide tree-ring network and calculated the difference between the daily and monthly approach. Day-wise aggregated correlations were on average higher by 0.071. In comparison to temperature data, the benefit of using daily data is greater for precipitation data.

The functionality of the daily analysis is based on a running window that simultaneously aggregates daily data and calculates correlations between proxy and aggregated daily data. The primary function of dendroTools is *daily_response()*, the basic functionality of which has already been presented by Jevšenak and Levanič (2018). Recently, new features were added to the package that extend the basic functionality and offer a variety of methods that could be useful for researchers from the dendroclimatological community and beyond.

The most important novelty are bootstrapped correlations, which enable the calculation of confidence intervals of correlation coefficients or (adjusted) explained variance. Partial correlations are commonly applied in dendroclimatology due to correlations between temperature and precipitation data (e.g. Marquardt et al., 2019; Zhang et al., 2014). A new function is available to effectively organize the required daily data format. Developed generic S3 plotting and summary functions (Chambers, 2014) provide effective methods for the interpretation of the calculated correlations. Finally, all functions that were primarily developed for daily data were also modified and now enable analyses using monthly data as well.

The purpose of this article is therefore to demonstrate the new features and functions in *dendroTools*, namely 1) data transformation, 2) bootstrapping, 3) partial correlation coefficients and 4) functions for analysis using monthly data. All examples presented below are coded in the R script *article_script.R*, which is given as supplementary material in executable format.

2. Installation and implementation

In this article, I refer to dendroTools v1.0.7, which is available under GNU General Public License, Version 3. The dendroTools R package is available from CRAN repository and can be installed with the standard command `>install.packages("dendroTools")`. Potential users are also invited to explore the current version under development, which is available from GitHub and can be installed with the command `>install_github("jernejjevsenak/dendroTools")`. To run the newest dendroTools, R version 3.4 or greater is needed. The current dendroTools relies on 18 other R packages.

Plotting is based on ggplot2 (Wickham, 2009), while data transformation is based on reshape2 (Wickham, 2007) and lubridate (Grolemund and Wickham, 2011).

3. Example data

The functionality of the new features in *dendroTools* is demonstrated using the freely available swit272 dataset (Bigler and Clalüna, 2012), which was downloaded from the International Tree-Ring Database (Zhao et al., 2019) and included in the *dendroTools* R package to make the examples presented here executable. The swit272 dataset is a standardized tree-ring width chronology of European larch (*Larix decidua*) from a high elevation site (2100 m) in southern Switzerland. The daily climate datasets used here are gridded E-OBS mean temperature and sums of precipitation (Cornes et al., 2018) on a 0.1 regular grid. These data have been available since 1950 and are also included in *dendroTools*. To load dendroTools and the data used for the examples presented here, type:

```
> library("dendroTools")
> data(swit272)
> data(swit272_daily_temperatures)
> data(swit272_daily_precipitation)
```

4. Transformation and quick preview daily data

Data preparation is an important step before analysing the relationships between daily data and a tree-ring proxy. The required format for daily data is a data frame with 366 columns and any number of rows, each representing one year, which is indicated as a row name. The common format of daily data provided by many online sources is a table with two columns, where the first column represents the date and the second is the value of the climate variable. To quickly transform such a format into a data frame with dimensions of 366 x n, dendroTools now offers the function *data_transform()*, whose functionality is based on functions from the *lubridate* R package (Grolemund and Wickham, 2011). The date can be in any of the listed formats in Table 1, but it must be correctly specified with the argument *date_format*. For example, if the date is in the format "1988-01-30" ("year-month-day"), the argument *data_format* must be "ymd". Daily temperature and precipitation data for swit272 chronologies are transformed with the following code:

```
> swit272_dt <- data_transform(swit272_daily_temperatures,
                                date_format = "ymd")
> swit272_dp <- data_transform(swit272_daily_precipitation,
                                date_format = "ymd")
```

Before the analysis of statistical relationships between daily data and a proxy record, it is recommended to quickly preview the daily data to check whether its values are reasonable and the number of missing values is not too large. To do so, use the function *glimpse_daily_data()*, which will plot the daily data and indicate all missing values. For the example data used in this article, missing values are indicated only for the end of the year 2019 (Figure 1). The temperature pattern shows higher summer and lower winter temperatures, while precipitation shows no obvious pattern, with many zeros and randomly distributed precipitation events.

```
> glimpse_precipitation <- glimpse_daily_data(swit272_dp)
> glimpse_temperatures <- glimpse_daily_data(swit272_dt)
```

Table 1: Examples of date formats with example and the appropriate *date_format* argument selection in *data_transform()*

Date format	**Example**	**Argument *date_format***
year-month-day	"1988-01-30"	"ymd"
year-day-month	"1988-30-01"	"ydm"
month-year-day	"01-1988-30"	"myd"
month-day-year	"01-30-1988"	"mdy"
day-year-month	"01-1988-30"	"dym"
day-month-year	"01-30-1988"	"dmy"

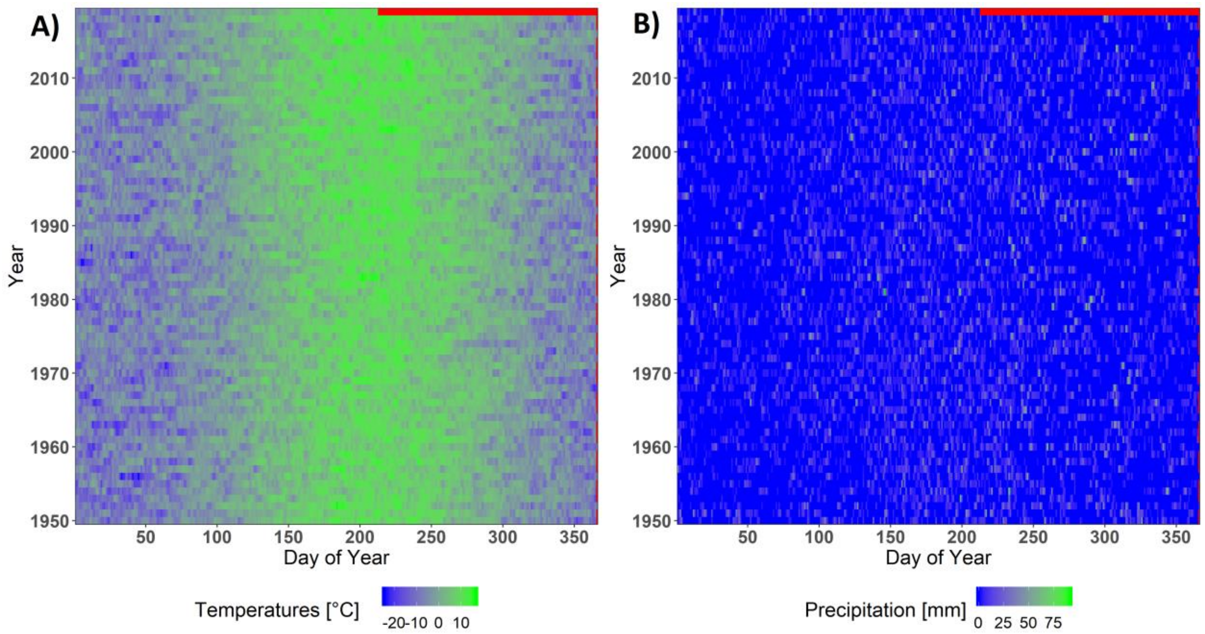

Figure 1: A quick preview of A) temperature and B) precipitation daily data obtained from *glimpse_daily_data()*.

5. Partial correlations from daily data

A partial correlation coefficient describes the strength of the linear relationship between two variables, holding constant a number of other variables (Freund et al., 2010). It is often used in dendroclimatological investigations to analyse the effect of temperature on a tree-ring parameter while at the same time controlling for the precipitation effect, or vice versa. This methodology was first implemented as the MATLAB program seascorr (Meko et al., 2011) and is now also available in the treeclim R package (Zang and Biondi, 2015) as the function *seascorr()*. Both implementations are available only for monthly climate observations.

Here, I present the same methodology that can be used on climate data on a daily scale and is implemented in the function *daily_response_seascorr()*. To analyse partial correlations, three data frames are needed: 1) a tree-ring proxy, 2) primary climate data and 3) secondary climate data for control. The tree-ring proxy must be organized as a data frame with one column representing proxy values, while years are indicated as row names. Primary climate data is assigned to the *env_data_primary* argument, while secondary climate data is assigned to *env_data_control*. The organization of daily climate data must be the same as described in the previous section. The range of analysis is controlled with *lower_limit* and *upper_limit* arguments. To consider all window widths between 21 and 270, set the *lower_limit* to 21 and *upper_limit* to 270. Daily data will be aggregated using all window widths between the lower and upper limits. Importantly, both limits are included in the considered window widths. The default measure of association is the Pearson correlation coefficient, but Kendall and Spearman correlation coefficients can also be used. This functionality is controlled with the *pcor_method* argument. I highly recommend using the feature of automatically sub setting data to only matching years. For example, the swit272 chronology spans from 1739 to 2011, while daily data are available only for the period from 1950 to 2019. If the argument *row_names_subset* is set to TRUE, the *daily_response_seascorr()* function will automatically subset the data to keep only matching years and provide results for the analysed period only, i.e. 1950 – 2011. The function *daily_response_seascorr()* is computationally expensive and takes several minutes to complete all calculations. To interpret the results, in addition to plotting methods, a generic S3 *summary()* function is now available. The result of *summary()* output is given in Table 2 and provides information on the attributes used in the analysis and, most importantly, calculated maximal partial correlation coefficient and described time window associated with the maximal correlation coefficient.

```
> pcor_results <- daily_response_seascorr(response = swit272,
                           env_data_primary = swit272_dt,
                           env_data_control = swit272_dp,
                           row_names_subset = TRUE,
                           lower_limit = 21, upper_limit = 270,
                           remove_insignificant = TRUE,
                           aggregate_function_env_data_primary = "mean",
                           aggregate_function_env_data_control = "sum",
                           alpha = 0.05, pcor_method = "spearman")
> summary(pcor_results)
```

Table 2: Output of the *summary()* function for the example of partial correlation analysis. The optimal climate signal is calculated for the period Jul 18 – Aug 8, which is from day 199 and the 22 following days. In this example, bootstrap was not used, and therefore the confidence interval is not given.

Variable	Value
approach	daily
method	Partial Correlation Coefficient (spearman)
metric	#N/A
analysed_years	1950 - 2011
maximal_calculated_metric	0.456
lower_ci	#N/A
upper_ci	#N/A
reference_window	Starting Day of Optimal Window Width: Day 199
analysed_previous_year	FALSE
optimal_time_window	Jul 18 - Aug 08
optimal_time_window_length	22

6. Bootstrapped correlation coefficients

The bootstrapping method is a computer-based method for assigning measures of accuracy to statistical estimates (Efron and Tibshirani, 1993). In the dendroTools R package, bootstrapping is available to estimate the confidence intervals of selected statistical metrics, i.e. correlation coefficient, explained variance or adjusted explained variance. To use bootstrap, set the argument *boot* as TRUE. The number of bootstrap samples is defined with the *boot_n* argument, while the confidence levels are specified with the *boot_conf_int* argument. In the following example, bootstrapped correlation coefficients are calculated with the *daily_response()* function for daily temperature records and swit272 chronology, while the bootstrap procedure is also available in the *daily_response_seascor()* and functions for the analysis based on monthly data. It must be noted that bootstrapping procedures are extremely time consuming. The example presented here took about 1.5 hours to complete the calculation of all bootstrapped correlations. To reduce the time needed for calculations, the amount of considered window widths should be reduced or, alternatively, the number of bootstrapped resamples lowered. However, such reductions might result in incomplete analysis. The optimal way for assessing the results is by using the *summary()* function (Table 3), while the upper and lower confidence intervals can be obtained manually by exploring the output list from the *daily_response()* function. To do so, type `boot_results$boot_lower` and `boot_results$boot_upper`.

```
> boot_results <- daily_response(response = swit272,
                              env_data = swit272_dt,
                              row_names_subset = TRUE,
                              lower_limit = 21, upper_limit = 270,
                              method = "cor",
                              cor_method = "pearson",
                              remove_insignificant = TRUE,
                              aggregate_function = "mean",
                              boot = TRUE, boot_n = 1000,
                              boot_conf_int = 0.95)
> summary(boot_results)
```

Table 3: Output of the *summary()* function for the example of bootstrapped correlation coefficients. The highest calculated correlation coefficient was 0.413 with lower and upper limits of 0.232 and 0.567.

Variable	Value
approach	daily
method	Correlation Coefficient (pearson)
metric	#N/A
analysed_years	1950 - 2011
maximal_calculated_metric	0.413
lower_ci	0.232
upper_ci	0.567
reference_window	Starting Day of Optimal Window Width: Day 170
analysed_previous_year	FALSE
optimal_time_window	Jun 19 - Aug 15
optimal_time_window_length	58

7. Analysis of climate-growth relationships using monthly data

Both the *daily_response()* and *daily_response_seascor()* functions also have variations that were developed to analyse climate-growth relationships using data on a monthly scale: *monthly_response()* and *monthly_response_seascor()*. The arguments in both function variations are very similar. Monthly data should be organized as a data frame with twelve columns (months), where each row represents one year. Years should be indicated as row names. Monthly data can be obtained from various online sources, but it is also possible to transform daily data into monthly with the *data_transform()* function (see below). In addition to the *format* argument, which must be set as *"monthly"*, the aggregation function should be specified. This could be *"mean"*, *"sum"* or *"auto"* (default). The last choice is based on the share of zeros in the data and, if the share of zeros is greater than 10 %, the function algorithm assumes precipitation data and aggregates values using the sum function, otherwise the algorithm assumes temperature data and aggregates values using the mean function. An example of *monthly_response()* is given below, where *pearson* correlations are analysed for monthly mean temperatures and swit272 chronology. To visualise results, a generic S3 *plot()* method is available (Figure 2).

```
> swit272_mt <- data_transform(swit272_daily_temperatures,
                               format = "monthly",
                               monthly_aggregate_function = "auto")
> monthly_results <- monthly_response(response = swit272,
                                 env_data = swit272_mt,
                                 row_names_subset = TRUE,
                                 lower_limit = 1, upper_limit = 12,
                                 remove_insignificant = FALSE,
                                 alpha = 0.5, method = "cor",
                                 aggregate_function = "mean",
                                 cor_method = "pearson")
> plot(monthly_results, type = 1)
> plot(monthly_results, type = 2)
```

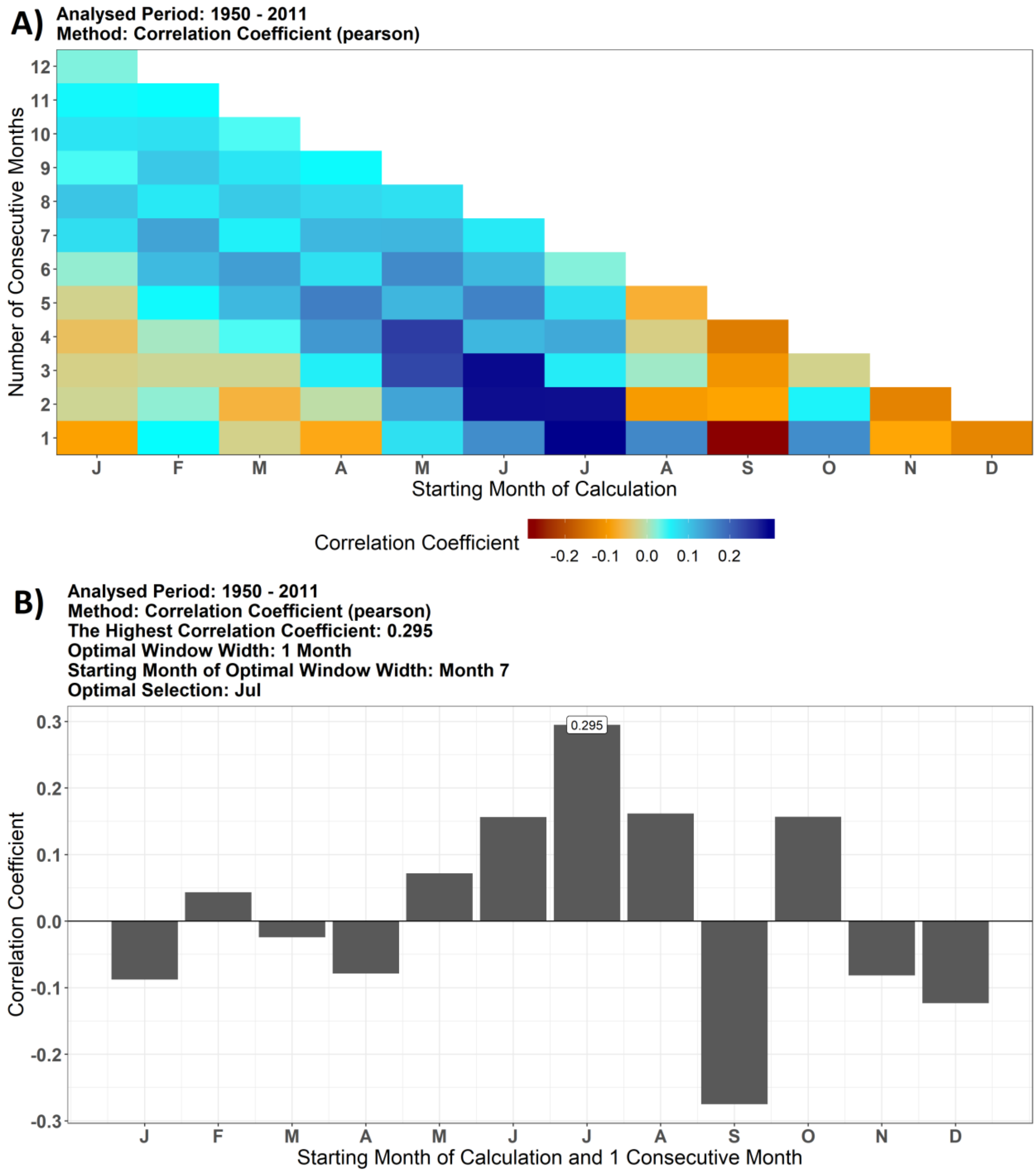

Figure 2: A) Heatmap of the temporal pattern of monthly climate-growth relationships and B) highlighted optimal window with the highest calculated correlation coefficient. Both figures show significant positive correlations with summer and significant negative correlations with September temperatures.

8. Conclusions

Due to the advantages related to the daily data approach, many authors have decided to calculate climate-growth correlations using daily data (e.g. Kaczka et al., 2018; Nechita et al., 2019). Arguably, the most evident disadvantage of the *daily_response()* and *daily_response_seascorr()* functions is the so-called problem of multiple testing, which increases type I error. However, it must be noted that while the multiple testing problem relates to situations where numerous independent statistical tests are applied simultaneously, in the dendroTools algorithms multiple tests are highly dependent due to the running window approach. In addition, p correction methods can result in increased risk of type II errors (Perneger, 1998). Therefore, no p adjustment method is implemented in the dendroTools functions, but users should be aware of this issue and rely mostly on highly significant correlations that are stable in time and biologically interpretable.

Acknowledgments

I acknowledge the E-OBS dataset from the EU-FP6 project UERRA (http://www.uerra.eu) and the Copernicus Climate Change Service, and the data providers in the ECA&D project (https://www.ecad.eu).

Funding

Funding for this study was provided by the Slovene Research Agency: Program and Research Group “Forest biology, ecology and technology” P4-0107 and basic research projects J4-8216 “Mortality of lowland oak forests - consequence of lowering underground water or climate change?”.

9. References

Beck, W., Sanders, T.G.M., Pofahl, U., 2013. CLIMTREG: Detecting temporal changes in climate–growth reactions – A computer program using intra-annual daily and yearly moving time intervals of variable width. Dendrochronologia 31, 232-241.

Bigler, C., Claluna, A., 2012. NOAA/WDS Paleoclimatology - Bigler - Sils-Maria GR Blais dal Fö - LADE - ITRDB SWIT272. https://doi.org/10.25921/37hm-5971.

Chambers, J.M., 2014. Object-oriented programming, functional programming and R. Statistical Science 29, 167-180.

Cornes, R.C., van der Schrier, G., van den Besselaar, E.J.M., Jones, P.D., 2018. An Ensemble Version of the E-OBS Temperature and Precipitation Data Sets. Geophys. Res. Lett.: Atmospheres 123, 9391-9409.

Efron, B., Tibshirani, R., 1993. An introduction to the bootstrap. Chapman & Hall, New York, xvi, 436 p. pp.

Freund, R.J., Wilson, W.J., Mohr, D.L., 2010. CHAPTER 8 - Multiple Regression, in: Freund, R.J., Wilson, W.J., Mohr, D.L. (Eds.), Statistical Methods (Third Edition). Academic Press, Boston, pp. 375-471.

Grolemund, G., Wickham, H., 2011. Dates and Times Made Easy with lubridate. J Stat Softw 40, 1-25.

Hulist, A., Janecka, K., Kaczka, R., 2016. DendroCorr the simple and powerful software to calculate the growth/climate response, in: TRACE-Tree Rings in Archaeology, Climatology and Ecology. GFZ German Research Centre for Geoscience, pp.

Jevšenak, J., 2019. Daily climate data reveal stronger climate-growth relationships for an extended European tree-ring network. Quat. Sci. Rev. 221, 105868.

Jevšenak, J., Levanič, T., 2018. *dendroTools*: R package for studying linear and nonlinear responses between tree-rings and daily environmental data. Dendrochronologia 48, 32−39.

Kaczka, R., Janecka, K., Czajka, B., Hulist, A., 2018. Application of daily climate data in dendroclimatic analyses. Case study for Norway spruce from the Tatras. Studia i Materiały Centrum Edukacji Przyrodniczo-Leśnej 20, 7-17.

Marquardt, P.E., Miranda, B.R., Jennings, S., Thurston, G., Telewski, F.W., 2019. Variable climate response differentiates the growth of Sky Island Ponderosa Pines. Trees 33, 317-332.

Meko, D.M., Touchan, R., Anchukaitis, K.J., 2011. Seascorr: A MATLAB program for identifying the seasonal climate signal in an annual tree-ring time series. Computers & Geosciences 37, 1234-1241.

Nechita, C., Macovei, I., Popa, I., Badea, O.N., Apostol, E.N., Eggertsson, Ó., 2019. Radial growth-based assessment of sites effects on pedunculate and greyish oak in southern Romania. Sci. Total Environ. 694, 133709.

Perneger, T.V., 1998. What's wrong with Bonferroni adjustments. BMJ 316, 1236-1238.

R Core Team, 2019. R: A language and environment for statistical computing. R Foundation for Statistical Computing, http://www.R-project.org/.

Wickham, H., 2007. Reshaping data with the reshape package. J Stat Softw 21, 1-20.

Wickham, H., 2009. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 213 pp.

Zang, C., Biondi, F., 2015. treeclim: an R package for the numerical calibration of proxy-climate relationships. Ecography 38, 431-436.

Zhang, Y., Shao, X.M., Yin, Z.Y., Wang, Y., 2014. Millennial minimum temperature variations in the Qilian Mountains, China: evidence from tree rings. Clim. Past 10, 1763-1778.

Zhao, S., Pederson, N., D'Orangeville, L., HilleRisLambers, J., Boose, E., Penone, C., Bauer, B., Jiang, Y., Manzanedo, Rubén D., 2019. The International Tree-Ring Data Bank (ITRDB) revisited: Data availability and global ecological representativity. J. Biogeogr. 46, 355-368.