Dendrochronologia 48 (2018) 32–39

Contents lists available at ScienceDirect

Dendrochronologia

journal homepage: www.elsevier.com/locate/dendro

# *dendroTools:* R package for studying linear and nonlinear responses between tree-rings and daily environmental data

Jernej Jevšenak, Tom Levanič*

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

A R T I C L E I N F O

*Keywords:*
Dendroclimatology
Daily climate data
Running window
Nonlinear modelling
Tree-ring proxies
Climate reconstruction

A B S T R A C T

We introduce in this paper the *dendroTools* R package for studying the statistical relationships between tree-ring parameters and daily environmental data. The core function of the package is *daily_response()*, which works by sliding a moving window through daily environmental data and calculating statistical metrics with one or more tree ring proxies. Possible metrics are correlation coefficient, coefficient of determination and adjusted coefficient of determination. In addition to linear regression, it is possible to use a nonlinear artificial neural network with the Bayesian regularization training algorithm (*brnn*). *dendroTools* provides the opportunity to use daily climate data and robust nonlinear functions for the analysis of climate-growth relationships. Models should thus be better adapted to the real (continuous) growth of trees and should gain in predictive capabilities. The *dendroTools* R package is freely available in the CRAN repository. The functionality of the package is demonstrated on two examples, one using a mean vessel area (MVA) chronology and one a traditional tree-ring width (TRW).

## 1. Introduction

R computer language (R Core Team, 2017) is one of the most powerful platforms for analysing tree-ring data. Many useful packages have been developed in recent decades that are freely available to the tree-ring community. The *dplR* package (Bunn, 2008, 2010) is widely used to perform several standard analyses, including interactive detrending, chronology building and the calculation of standard descriptive statistics, and is slowly replacing the traditional software for tree-ring standardisation ARSTAN. The R package *treeclim* (Zang and Biondi, 2015) provides a unified and fast compilation of established methods, while adding novel functions, such as static and moving bootstrapped response and correlation functions, seasonal correlation analysis, a test for spurious temporal changes in proxy-climate relations, and the evaluation of reconstruction skills. Some other useful R packages developed for tree-ring analysis are *dendrometeR* (van der Maaten et al., 2016), *CAVIAR* (Rathgeber et al., 2011), *pointRes* (van der Maaten-Theunissen et al., 2015), *measuRing* (Lara et al., 2015), *TRADER* (Altman et al., 2014) and tracheideR (Campelo et al., 2016). These R packages are of significant importance and provide the opportunity to analyse tree-ring data more effectively. In addition to R packages, there are also other types of software that are commonly used for identifying climate signal in an annual tree-ring time series. Two of them are Seascorr (Meko et al., 2011), which runs in MATLAB; and DENDROCLIM2002 (Biondi and Waikul, 2004), a C++ programme.

The CLIMTREG programme was developed by Beck et al. (2013) and provides the possibility of calculating climate–growth correlations based on daily climate data using variable temporal width together with moving correlations to accommodate short term as well as long term influences. The programme has been used in several studies (e.g. Liang et al., 2013; Castagneri et al., 2015), but unfortunately has not been further developed, since the company that produced the Gfa-Basic32 programming language no longer exists. Despite the great potential of improving understanding of the climate-growth relationship, there is currently no similar function available in R. The identified methodological gap could be filled by our newly developed R package *dendroTools* (Jevšenak and Levanič, 2018), especially with its core function *daily_response()*. This function provides the possibility of analysing linear and nonlinear relationships between tree-ring and daily environmental data, and could therefore be important to help researchers identify tree-climate relationships. With the proposed methodology, models should be better adapted to the real (continuous) growth of trees and should gain predictive capabilities, which should result in more accurate climate reconstructions and better understanding of climate-growth relationships.

Common practice in dendroclimatology is to correlate one or more tree-ring proxies (predictors) to monthly or seasonal climate data (predictands). By using monthly data, some climate signal is inevitably

* Corresponding author.
*E-mail address:* tom.levanic@gozdis.si (T. Levanič).

https://doi.org/10.1016/j.dendro.2018.01.005
Received 25 October 2017; Received in revised form 22 January 2018; Accepted 24 January 2018
Available online 31 January 2018
1125-7865/ 

lost, mainly because months are invented categories not based on any of the laws of nature. Growth is a continuous process and should not be limited by artificially set monthly borders. With the *daily_response()* function from the *dendroTools* R package, temporal changes in climate-growth response are analysed and results can be later used for various dendroclimatological applications. It is not new for daily environmental data to be used in combination with tree-ring proxies. The process-based Vaganov-Shashkin model uses daily temperature and precipitation data to simulate tree-ring chronologies (e.g. Touchan et al., 2012). Chun et al. (2017) used tree-ring width information to improve daily-scale reconstructions of rainfall extremes.

The aim of this article is to present the functionality of the *dendroTools* R package, with an emphasis on the *daily_response()* function. Two case studies have been used to do so, one using a mean vessel area (MVA) and one using a tree-ring width (TRW) parameter.

## 2. *dendroTools* description

### 2.1. Package requirements, installation and dependences

The *dendroTools* R package will run on R version 3.4 or higher, simply because it depends on certain other packages that do not work in older versions of R. After installing the right version of R, *dendroTools* can be installed from the Comprehensive R Archive Network (CRAN) with the following command: install.packages("dendroTools") and loaded with: library ("dendroTools"). The current version (0.0.5) relies on 15 other R packages. Those that are important for the functionality of the *daily_response()* function are: *ggplot2* (Wickham, 2009), *oce* (Kelley and Richards, 2017), *brnn* (Pérez-Rodríguez and Gianola, 2016), *reshape2* (Wickham, 2007), *scales* (Wickham, 2016), *stats* (R Core Team, 2017), *reshape* (Wickham, 2007), *MLmetrics* (Yan, 2016), *dplyr* (Wickham et al., 2017) and dcv (Li and Zhang, 2010). In addition, R users should have installed the appropriate Java, i.e., 32-bit Java for 32-bit R and 64-bit Java for the 64-bit R version.

### 2.2. Package functionality

The *daily_response()* function is the core function of the *dendroTools* R package. Although the name of this function suggests the connection to the response functions presented by Fritts (1976), this is not the case; they are two different concepts. The main purpose of the *daily_response ()* is to analyse temporal changes of relationships between tree-ring proxies and daily environmental data. The function calculates all possible statistical metrics between different ranges of daily data and one or more response variables. The key purpose is to find the optimal consecutive sequence of days that are linearly or nonlinearly related to one or more response variable (i.e., tree-ring proxies).

The function *daily_response()* works by sliding a moving window through daily environmental data, aggregating daily environmental data within each window and calculating its averages (Fig. 1A), which are then used to calculate the selected statistical metric – i.e., correlation coefficient, coefficient of determination or adjusted coefficient of determination (Fig. 1B). Two data frames have to be passed to *daily_response()*, i.e., *response* and *env_data*. *response* are data frames with one or more tree-ring proxy variables. Rows represent years and columns represent proxy variables. Years should be included as row names of a data frame to avoid errors. *env_data* is a data frame with daily environmental data (e.g., temperature, precipitation or similar). Rows represent years and columns represent a day of a year, starting with day 1 of the year in column 1. Years should be included as row names of a data frame. The examples of *response* and *env_data* are given in Table 1.

To use *daily_response()*, the user should first decide whether to use a fixed or progressive window for calculations of moving averages. To use a fixed window, select its width by assigning an integer to the argument *fixed_width*. When the user is interested in many different windows, *lower_limit* and *upper_limit* arguments are available. In this case, all window widths between the lower and upper limits will be considered. In this context, window width representative of a specific day of the year (DOY) is defined as the values for this particular day and the number of subsequent days corresponding to window width. All calculated metrics are stored in a matrix (Fig. 1C). This matrix is available

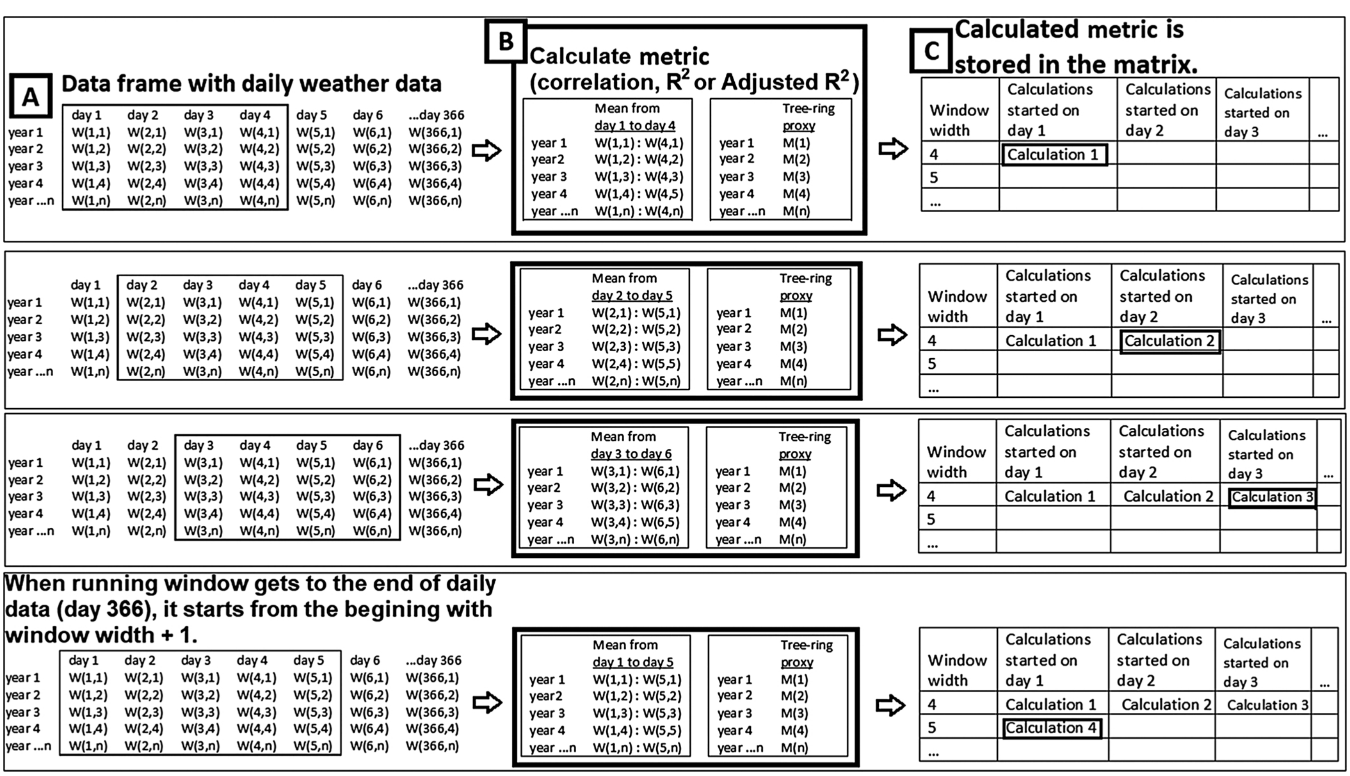


**Fig. 1.** Schematic presentation of the running window of the *daily_response()* function. In this example, the initial window width is set to 4.

**Table 1**
Required data frame organization of the *response* (left and middle table) and *env_data* (right table) inputs for the *daily_response()* function. Years should be included as row names of data frames.

| | MVA |
|---|---|
| 1961 | 7.18567 |
| 1962 | 5.59846 |
| 1963 | 5.87261 |
| 1964 | 6.50313 |
| 1965 | 5.66054 |
| 1966 | 6.00276 |
| 1967 | 6.01883 |
| 1968 | 7.36647 |
| 1969 | 5.71727 |
| 1970 | 5.98721 |
| 1971 | 6.07254 |
| 1972 | 5.87815 |
| 1973 | 5.13292 |
| 1974 | 6.26117 |
| 1975 | 5.74098 |
| 1976 | 5.75330 |
| 1977 | 5.93055 |
| 1978 | 5.52767 |
| 1979 | 5.52998 |

| | TRW |
|---|---|
| 1757 | 1.392 |
| 1758 | 1.130 |
| 1759 | 1.483 |
| 1760 | 1.183 |
| 1761 | 1.256 |
| 1762 | 1.146 |
| 1763 | 1.440 |
| 1764 | 1.209 |
| 1765 | 0.854 |
| 1766 | 0.614 |
| 1767 | 0.677 |
| 1768 | 0.602 |
| 1769 | 0.875 |
| 1770 | 0.559 |
| 1771 | 0.578 |
| 1772 | 0.541 |
| 1773 | 0.631 |
| 1774 | 0.773 |
| 1775 | 1.171 |

| | X1 | X2 | X3 | X4 | … | X365 | X366 |
|---|---|---|---|---|---|---|---|
| 1961 | -0.1 | 0.5 | 1.9 | 2.5 | … | 4.3 | *NA* |
| 1962 | 4.7 | 6.5 | -1.1 | -3.3 | … | 1.4 | *NA* |
| 1963 | -0.2 | -0.2 | 0.8 | 1.5 | … | -4.3 | *NA* |
| 1964 | -4.9 | -5 | -5.3 | -5.6 | … | -3.9 | -7.8 |
| 1965 | -0.8 | 2.1 | 0.9 | -1.8 | … | 0.4 | *NA* |
| 1966 | -1.4 | 0.3 | 2 | 1.2 | … | 2 | *NA* |
| 1967 | -1 | 0.8 | -0.4 | -3.5 | … | 0.2 | *NA* |
| 1968 | -1.2 | -3.5 | -9.4 | -8.4 | … | -11.1 | -10.3 |
| 1969 | -12.1 | -8.6 | -3.8 | -2.3 | … | -1.6 | *NA* |
| 1970 | -1.3 | -3.1 | -1.1 | 2.1 | … | 0.1 | *NA* |
| 1971 | -3.5 | -6.5 | -7.8 | -9.9 | … | 0.9 | *NA* |
| 1972 | 0.8 | 0.8 | 0.4 | 1.1 | … | -4.6 | -4.3 |
| 1973 | -1.6 | -0.2 | 1.1 | 0.8 | … | 0.2 | *NA* |
| 1974 | 0.2 | 0.4 | 1 | 1.3 | … | 1.9 | *NA* |
| 1975 | 0.9 | 1.1 | 1.7 | -2.7 | … | -3.8 | *NA* |
| 1976 | -1.2 | 4.3 | 2.7 | 3.8 | … | -8.7 | -8.5 |
| 1977 | 4.6 | 2.1 | 1.7 | 0.5 | … | 2.1 | *NA* |
| 1978 | 1.7 | 3.4 | 5.8 | 5.3 | … | 8.2 | *NA* |
| 1979 | 1.1 | -8.2 | -10.6 | -7.8 | … | -0.6 | *NA* |

as the first element of the output list of the *daily_response()* function. The optimal window (i.e., optimal consecutive sequence of days) is then found, which returns the highest calculated metric. For a full description of all the other arguments, including examples, see the *dendroTools* manual at https://cran.r-project.org/web/packages/dendroTools/dendroTools.pdf. The output of the *daily_reponse()* function is a list with 13 elements (see Table 2), which can be retrieved by calling their names, as demonstrated in later examples

### 2.3. Nonlinear brnn function

The *daily_response()* function enables linear and nonlinear climate-tree analysis. As a nonlinear method, an artificial neural network with a Bayesian regularization (*brnn*) training algorithm is implemented. This method is used because 1) it has already been successfully applied to tree-ring data by Jevšenak and Levanič (2016), 2) it is robust to overfitting, 3) it is easy to use and 4) it usually produces a sigmoid shaped function between tree-ring parameter and climate data, which should in theory be a better fit to tree-climate data. The *brnn* model in R can be fitted with the *brnn* R package (Pérez-Rodríguez and Gianola, 2016). A simple code is needed, such as brnn_model < − brnn(y ~ x, data = data, neurons = 1). The only tuning parameter needed is *neurons*. In dendroclimatological models with 1 independent variable, this argument should be between 1 and 3.

**Table 2**
The description of the output list elements of the *daily_response()* function.

| Element name | Element description |
|---|---|
| $calculations | a matrix with calculated metrics |
| $method | the character string of a method |
| $metric | the character string indicating the metric used for calculations |
| $analysed_period | the character string specifying the analysed years |
| $optimized_return | data frame of aggregated (averaged) daily data that return the highest metric |
| $optimized_return_all | a data frame with aggregated daily data that returned the optimal result for the entire *env_data* (and not only subset of analysed years) |
| $transfer_function | a scatter plot and transfer function of optimized return and response data |
| $cross_validation | a data frame with cross validation results |
| $temporal_stability | a data frame with calculations of selected metric for different temporal subsets |
| $plot_heatmap | ggplot2 object: a heatmap of calculated metrics |
| $plot_extreme | ggplot2 object: line plot of a row with the highest value in a matrix of calculated metrics |
| $plot_specific | ggplot2 object: line plot of a row with a selected window width in a matrix of calculated metrics |
| $PCA_output | *princomp* object: the result output of the PCA analysis |

Briefly, the *brnn* function fits a two-layer neural network as described by Mackay (1992) and Foresee et al. (1997). It uses the algorithm introduced by Nguyen and Widrow (1990) to assign initial weights and the Gauss-Newton algorithm to perform the optimization. For a full description, including a mathematical derivation of the *brnn* algorithm, see Perez-Rodriguez et al. (2013). The biggest disadvantage related to this black box principle is that there are no coefficients with confidence intervals to estimate the uncertainty related to predictions.

## 3. Examples of workflow

### 3.1. Example data

Two examples are used to demonstrate the use of our method of studying the relationship between tree-ring parameters and daily temperatures. For *example_MVA*, we try to identify correlations between the mean vessel area (MVA) parameter of *Quercus robur* and daily mean temperature data for the meteorological station Ljubljana. Six trees for wood-anatomical analysis were cored from a lowland forest in fall 2012. For more information about the site and chronology characteristic, see Jevšenak and Levanič (2015). In *example_TRW*, similarly, the tree-ring width (TRW) parameter of *Picea abies* is used to find the optimal sequence of consecutive days that maximizes the climate signal. The TRW chronology represents Alpine forest and was downloaded from the National Centre for Environmental Information (https://www.ncdc.noaa.gov/). For more information about the TRW chronology, see Schweingruber (1981), The climate data used for *example_TRW* is the mean daily temperature for the meteorological station Kredarica. Climate data for our study was downloaded from KNMI Climate Explorer (https://climexp.knmi.nl). All datasets used in this paper are included in the *dendroTools* R package and can be obtained with the function *data ()*. Some additional information about the data for the two examples is given in Table 3.

### 3.2. Example_MVA

Data for *example_MVA* is saved in the data frame designated *data_MVA*. Daily data for the meteorological station Ljubljana is saved in the data frame called *LJ_daily_temperatures*. For *example_MVA*, simple

**Table 3**
General information about the data used for examples 1 and 2.

| | Tree-ring parameter | Species | Analysed period | Location | Elevation | Daily climate data |
|---|---|---|---|---|---|---|
| example_MVA | MVA (raw) | *Quercus robur* | 2012–1940 | Mlace (Lat: 46.3, Long: 15.51) | 300 m | Ljubljana (Lat: 46.06, Long: 14.51) |
| example_TRW | TRW (std) | *Picea abies* | 1955–1981 | Vršič (Lat: 46.47, Long: 13.76) | 1600 m | Kredarica (Lat: 46.38, Long: 13.85) |

running correlations will be used to find the optimal sequence of consecutive days. All possible window widths between 21 and 270 days, including the previous year, will be considered. The latter is achieved by setting the *previous_year* argument to *TRUE*. Specifically, we are interested in temporal changes of correlations for a window width of 90 days, so the parameter *plot_specific_window* is set to 90. For *example_MVA*, the *row_names_subset* argument is set to *TRUE*. This argument is particularly useful and allows the use of data frames of *response* and *env_data* with different years, i.e., a different number of rows, such as in Table 1. If *row_names_subset* is set to *TRUE*, the algorithm will automatically subset both data frames (i.e., environmental and tree-ring data) and keep only matching years, which will be used for calculations. To use this feature, years must be included as row names. There are many ways of doing this but there is also a *years_to_rownames()* function available in the *dendroTools* package. For *example_MVA*, all insignificant correlations were removed by setting the argument *remove_insignificant* to *TRUE*. The threshold for significance is set with the *alpha* argument. The method of assessing the temporal stability (*temporal_stability_check*) of correlations is set to "progressive". The progressive method splits data into *k* parts, calculates metric for the first part and then progressively adds 1 part at a time and calculates selected *metric*.

```
> library(dendroTools)
> data(data_MVA)
> data(LJ_daily_temperatures)
> example_MVA <- daily_response(response = data_MVA, env_data =
LJ_daily_temperatures, method = "cor", lower_limit = 21, upper_limit = 270,
row_names_subset = TRUE, previous_year = TRUE, remove_insignificant = TRUE,
alpha = 0.05, plot_specific_window = 90, temporal_stability_check =
"progressive", k = 5)
> example_MVA$plot_extreme
> example_MVA$plot_heatmap
> example_MVA$plot_specific
> example_MVA$temporal_stability
```

Results for *example_MVA* are visualised by retrieving the elements of the output list. The optimal sequence of consecutive days is visualised by calling example_MVA$plot_extreme (Fig. 2A). This feature explores the matrix of calculated metrics, finds the window width with the highest calculated metric, graphs it and indicates the sequence of days that returns the highest calculated metric. In titles, there is information about the analysed period, maximum correlation coefficient and optimum window width. The highest correlation coefficient, 0.77, was calculated with a window width of 59 days, starting on DOY 74 of the current growing season. The MVA parameter from the analysed site therefore contains the optimal climate signal from March 15 (DOY 74) to May 12 (DOY 132). This calculation is consistent with the study of xylogenesis in oak from a nearby site (Gričar, 2010), which reported that the period of most intense xylem cell production was assessed to be in the period April–May.

The average temperature from March 15 to May 12 for the analysed period is saved as a data frame – the fifth element of the output list. It can be retrieved by typing example_MVA$optimized_return. This data frame is used to calculate the temporal stability (example_MVA$temporal_stability) of correlation coefficients. The calculated values for different periods show that correlations are stable in time (Table 4).

Temporal changes of correlations for different window widths were visualised by typing example_MVA$plot_heatmap (Fig. 2B). The highest correlations were calculated for DOY around 440, with window width between 40 and 70. Note the temporal patterns, i.e. clear vertical and diagonal structures. These are discussed later in the section, Caveats and limitations of the *daily_response()* function. To visualise the temporal correlations of a pre-defined window width of 90 days (Fig. 2C), type example_MVA$plot_specific. This window width shows a similar influence of temperatures from previous and current growing seasons.

### 3.3. Example_TRW

TRW data for *example_TRW* is saved in the data frame designated *data_TRW*. Daily data for the meteorological station Kredarica is saved in the data frame called *KRE_daily_temperatures*. In this example, the metric coefficient of determination is calculated using linear (*method = "lm"*) and nonlinear (*method = "brnn"*) methods. All possible window widths are considered between 21 days (three weeks) and 270 days (9 months).

```
> library(dendroTools)
> data(data_TRW)
> data(KRE_daily_temperatures)
>  example_TRW_lm  <-  daily_response(response  =  data_TRW,  env_data  =
KRE_daily_temperatures, method = "lm", metric = "r.squared", lower_limit =
21, upper_limit = 270, row_names_subset = TRUE)
> example_TRW_lm$plot_extreme
> example_TRW_lm$plot_heatmap

>  example_TRW_brnn  <-  daily_response(response  =  data_TRW,  env_data  =
KRE_daily_temperatures, method = "brnn", metric = "r.squared", lower_limit
= 21, upper_limit = 270,row_names_subset = TRUE)
> example_TRW_brnn$plot_extreme
> example_TRW_brnn$plot_heatmap
```

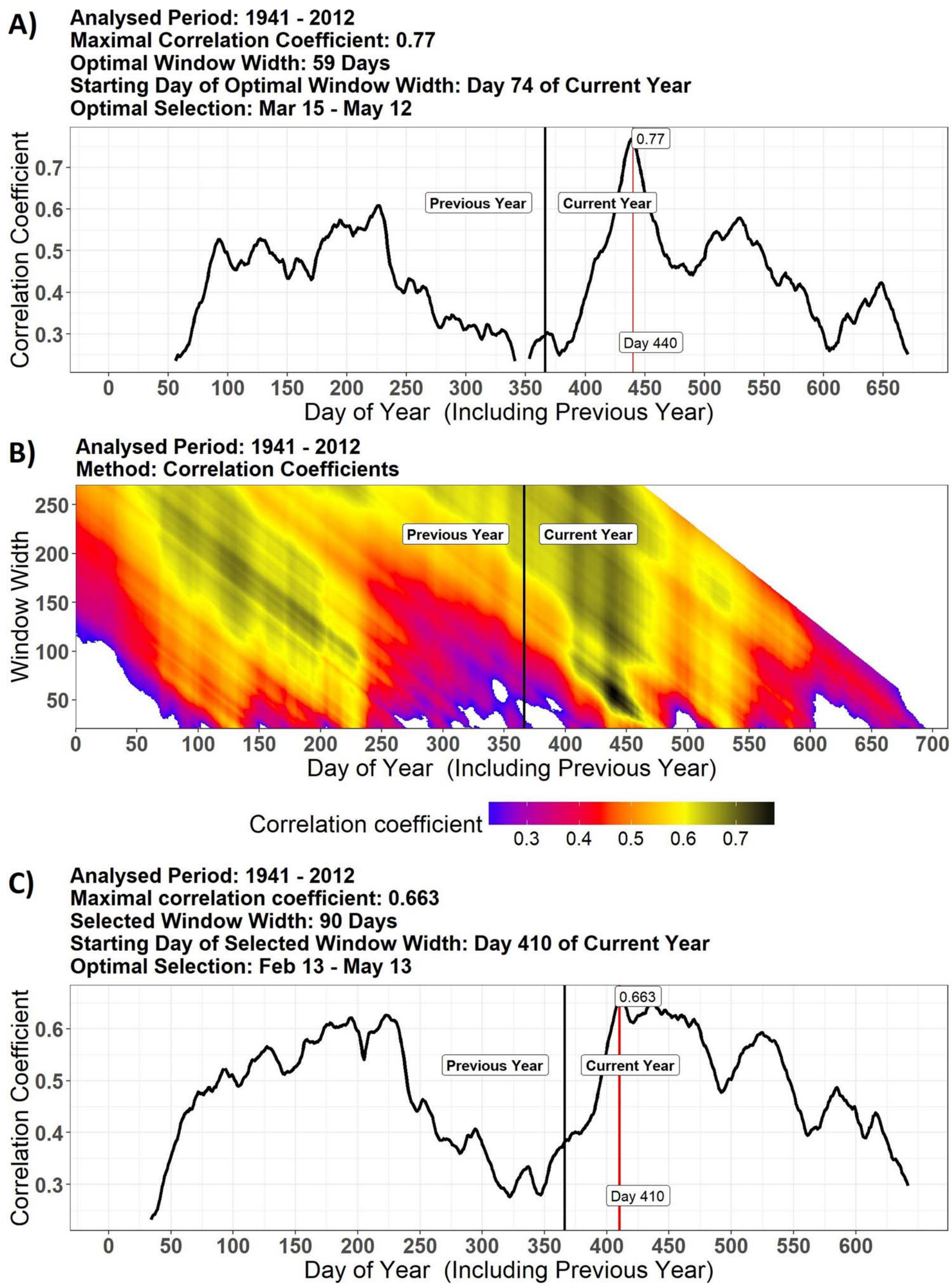


**Fig. 2.** Results for *example_MVA*: A) the maximised correlation coefficient, B) temporal patterns of climate-growth relationship and C) plot for a specific window width of 90 days. DOY on the x axis represents starting DOY and subsequent days of the respective window width. The broken line for A) and C) and white areas for B) are due to the removal of insignificant calculations (*remove_insignificant* argument in the *daily_response()* was set to *TRUE*).

**Table 4**
Temporal stability of correlation coefficients for the *example_MVA*.

| | Period | Correlation |
|---|---|---|
| 1 | 1941–1955 | 0.615 |
| 2 | 1941–1969 | 0.760 |
| 3 | 1941–1983 | 0.654 |
| 4 | 1941–1997 | 0.682 |
| 5 | 1941–2012 | 0.770 |

To visualise the optimal sequence of consecutive days, type example_TRW_lm$plot_extreme (Fig. 3A) and example_TRW_brnn$plot_extreme (Fig. 3D). Both linear and nonlinear algorithms suggest an optimal window starting on May 15 (DOY 135), with a span of 44 days (DOY 179, June 28). The highest calculated coefficient of determination with a linear algorithm (0.362) is slightly better than the coefficient of determination calculated with a nonlinear *brnn* algorithm (0.348). The optimal window width is in accordance with the typical growing season of conifers in the Alpine region close to the tree line. Rossi et al. (2007) reported the growing season of *Larix decidua*, *Picea abies* and *Pinus cembra* to be from May to July-August. Similarly, Swidrak et al. (2011) reported the onset and maximum growth rate of *Pinus cembra* from Eastern Alps to be on April 27 and June 23, respectively.

Temporal patterns of coefficients of determination are visualised by typing example_TRW_lm$plot_heatmap (Fig. 3B) and example_TRW_brnn$plot_heatmap (Fig. 3E). Again, both heatmaps show a similar pattern, with significant correlations only in late spring and summer with window widths lower than 150 days. Transfer functions of both algorithms show the relationship between the inputs and outputs.

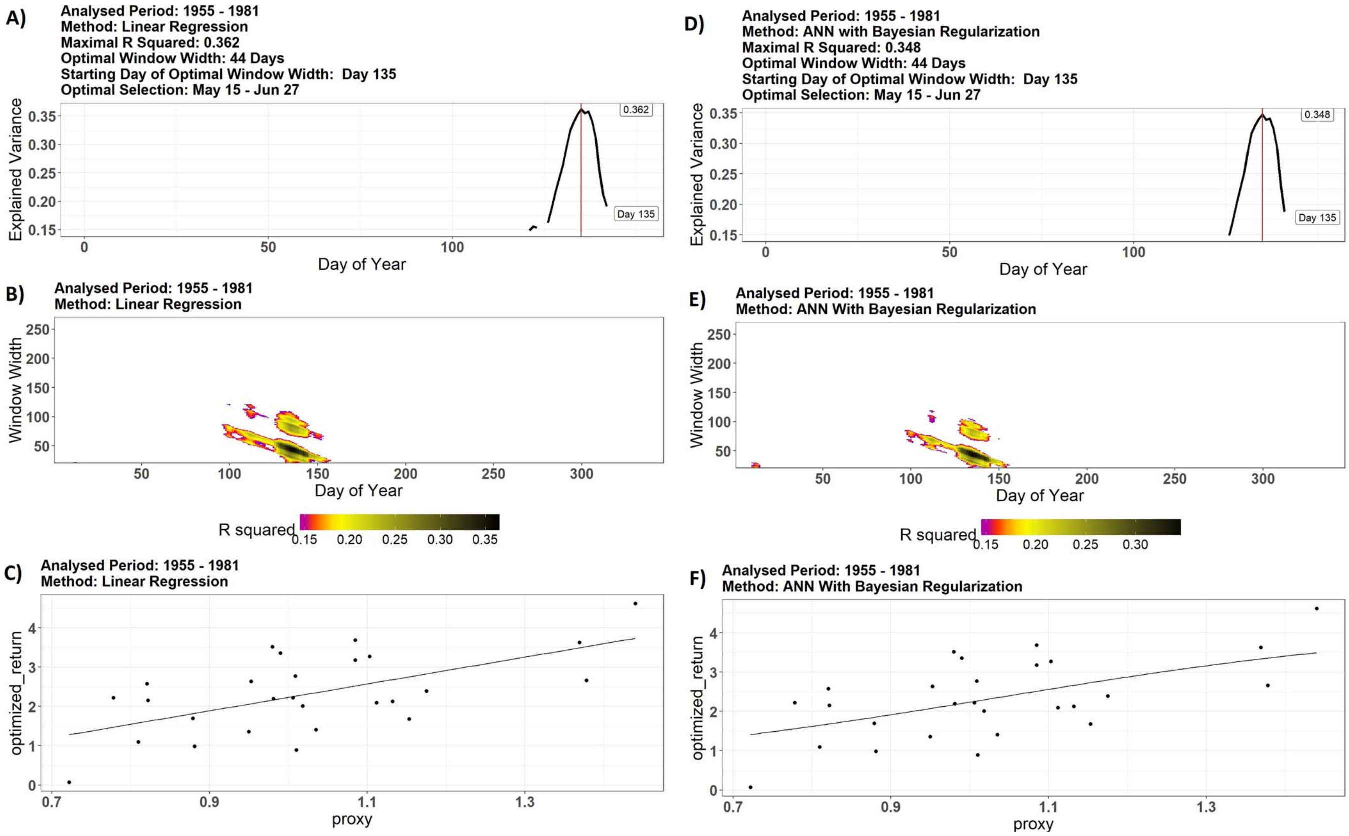


**Fig. 3.** Results for *example_TRW*: A) and D) maximised coefficient of determination, B) and E) temporal patterns of climate-growth relationship and C) and F) transfer functions for the *lm* and *brnn* models, respectively. DOY on the x axis represents starting DOY and subsequent days of the respective window width.

The two transfer functions are visualised by typing example_TRW_lm $transfer_function (Fig. 3C) and example_TRW_brnn$transfer_function (Fig. 3F). The two transfer functions assume a similar relationship between TRW and average temperature from May 15–June 28. However, the differences are greater for predictions close to the edges of the calibration data.

### 3.4. From daily_response() to climate reconstruction

Climate reconstruction is one of the most widely used applications in dendroclimatology. We therefore provide here an example of R code; how to use the output list of the *daily_response()* for t*example_TRW* to reconstruct climate with the *lm* and *brnn* functions. Aggregated daily data (*i.e.,* optimum selection) is stored as an element in the output list ($optimized_return) and can be used directly to calibrate models for climate reconstruction.

First, linear and *brnn* models are calibrated by using the $optimized_return data frame, and then used to reconstruct (*predict*) climate for the past period. Reconstructed temperatures are given in Fig. 4. The two reconstructions are similar; however, linear reconstruction provides more extreme predictions. These differences in reconstructed temperatures are directly related to differences between *lm* and *brnn* transfer functions (Fig. 3C and F). The linear transfer function assumes that the effect of temperatures on TRW is the same for the whole spectrum of temperatures. On the other hand, the *brnn* function assumes a different (more moderate) effect of temperatures for extreme conditions.

## 4. Caveats and limitations of the *daily_response()*

Our methodology is not robust enough to identify spurious correlations that may arise due to coincidence, autocorrelation etc. There are

```
> linear_model <- lm(Optimized_return ~ TRW, data =
example_TRW_lm$optimized_return)
> library(brnn)

>    brnn_model    <-    brnn(Optimized_return    ~    TRW,    data    =
example_TRW_brnn$optimized_return, neurons = 1)

>  lm_reconstruction  <-  data.frame(predictions  =  predict(linear_model,
newdata = data_TRW))
>  brnn_reconstruction  <-  data.frame(predictions  =  predict(brnn_model,
newdata = data_TRW))

> plot(x = row.names(data_TRW), y = lm_reconstruction$predictions, col =
"red", type = "l", xlab = "Year", ylab = "Average temperature May 15 - June
27 [°C]", cex.lab = 1.5, cex.axis = 1.5)
> lines(x = row.names(data_TRW), y = brnn_reconstruction$predictions, lty =
3, col = "blue")
>   legend(1915,   0.75,   legend   =   c("linear   reconstruction",   "brnn
reconstruction"), lty =c(1, 3), col = c("red", "blue"), cex = 1.2)
```

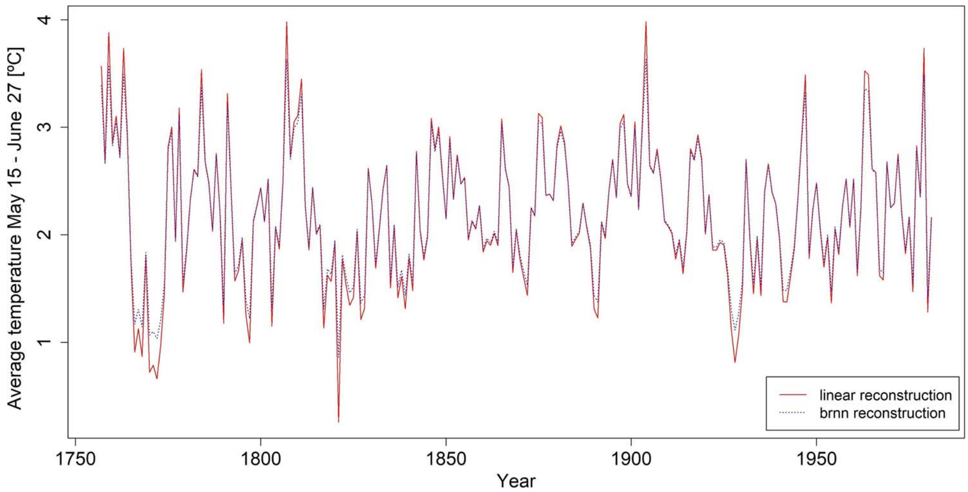


**Fig. 4.** Linear and nonlinear *brnn* climate reconstruction for the *example_TRW*.

patterns in Fig. 2B, i.e., clear vertical and diagonal (from top left to bottom right) structures. The vertical lines suggest two things. First of all, the sometimes abrupt colour change from one day to another suggests influential outliers, i.e., at a particular DOY, the average over the window will abruptly change either because a specific value is now included or another is left out. Secondly, the vertical lines show that specific windows that show a strong correlation (e.g., the windows around DOY 440) will indicate strong correlations for this DOY for most of the window sizes, despite the fact that some of these window sizes will include periods that, on a shorter window-scale, expressed low correlations or even insignificant correlations (as indicated by the diagonal lines, which represent the 'later' representation of this window but with shorter window sizes). As an example, the correlation for window size 250 for the period around DOY 440 is in the order of 0.7 but includes a period around DOY 650 with correlations lower than 0.4. It would not therefore be meaningful to choose this particular window and period, but for another data set and other specifications (range of window sizes) this may coincidentally turn out to be the highest correlation. Another feature of the diagonal lines is that they clearly show that the correlations abruptly change depending on the window size. Some of these issues may be compensated for by using median instead of mean. To do so, set the argument *use_median* to *TRUE*. However, median is less affected by very hot/cold temperatures and might therefore diminish correlations between *response* and *env_data*. All users of our tool should make their final selection of window size and period carefully.

In terms of window widths, we recommend not to select too small window sizes, since the likelihood of obtaining spurious correlations for small window widths may be comparatively higher, since small window sizes will incorporate more high-frequency variations, which may coincidentally match the proxy variations. In addition, selecting a window width that exceeds the period of the growing season may also result in some spurious correlations. However, if the selected window size is less than 14 (2 weeks) or greater than 270 (9 months), a warning is given but calculations will be performed anyway. Users should therefore select window sizes reasonably.

The *daily_response()* function does not address the risks that arise from repeating multiple significance tests simultaneously. For the *example_MVA* and *example_TRW*, 55375 calculations were needed to find the optimal sequence of consecutive days, so the use of any kind of *p* correction method would result in a very low number of significant correlations. With no correction, the chance of finding one or more significant correlations by chance alone is high. For our two examples, around 2700 calculations theoretically results in a type I error. Potential users should note this risk and set the threshold of significant correlations below 0.05 to reduce the likelihood of a type I error.

There is no special treatment for leap years, users should decide how to organize the *env_data*. February 29 of non-leap years could therefore be skipped, assigned *NA*, modelled as an average of values on February 28 and March 1, or similarly. In the examples used in this paper, February 29 of non-leap years was skipped, so those years had 365 days, while leap years had 366 days. However, users should note the small difference between various treatments and interpret results accordingly. The dates indicated by plotting methods in our examples (Figs. Figure 2A, Figure 2C and Figure 3A and D) are based on dates from a non-leap year, so there is no February 29 included.

Finally, *daily_response()* allows for including multiple tree-ring proxies simultaneously as potential independent variables for daily environmental data. However, users should select multiple proxies carefully and with caution, since there is nothing to prevent the inclusion of colinear variables. Including several proxies will result in higher explained variance but at the cost of degrees of freedom. In such cases, users should use the adjusted coefficient of determination and check the cross-validation results (e.g., example_MVA$cross_validation). If metrics on validation data are much lower than in calibration data, there is a problem of overfitting and users should exclude some proxies and repeat the analysis

## 5. Conclusions

The approach to analysing the relationship between daily data and tree-ring proxies with the *dendroTools* R package was introduced using two examples, one using MVA and one using TRW data. With the *daily_response()* function, the optimal sequence of consecutive days that is linearly or non-linearly related to a response variable can easily be found. As expected, TRW was related to late spring and early summer temperatures, while MVA corresponds to early spring temperatures.

The *daily_response()* function is a conceptually simple method and easy to use. It has many potential applications. The application of climate reconstruction is given for *example_TRW*. Climate changes affect tree-growth and, using our method, changes in the optimal window between past and present can also be analysed. It is also possible to run PC regression within the *daily_response()* function. For examples of the above mentioned applications, see the on-line vignette for the dendroTools R package (https://cran.r-project.org/web/packages/dendroTools/vignettes/Examples_daily_response.html).

The future development of the *dendroTools* package will be focused on improvement of the functionality of current functions and the implementation of new ones. One of them is *compare_methods()*, which effectively compares several regression methods and proposes the most suitable one. However, this function is not yet fully developed and is therefore not presented in this paper.

## Acknowledgment

The development of the *dendroTools* package was supported by the following programmes and projects financed by the Slovenian Research Agency: Program and Research Group "Forest biology, ecology and technology" P4-0107, basic research projects J4-5519 "Paleoclimate data enhances drought prediction in the W Balkan Region" and J4-8216 "Mortality of lowland oak forests – consequence of lowering underground water or climate change?" We are grateful for the valuable suggestions of the three anonymous reviewers, who also contributed to the better functionality of the *dendroTools* R package.

## References


Altman, J., Fibich, P., Dolezal, J., Aakala, T., 2014. TRADER: a package for tree ring analysis of disturbance events in R. Dendrochronologia 32, 107–112.

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.

Biondi, F., Waikul, K., 2004. DENDROCLIM2002: a C++ program for statistical calibration of climate signals in tree-ring chronologies. Comp. Geosci. 30, 303–311.

Bunn, A.G., 2008. A dendrochronology program library in R (dplR). Dendrochronologia 26, 115–124.

Bunn, A.G., 2010. Statistical and visual crossdating in R using the dplR library. Dendrochronologia 28, 251–258.

Campelo, F., Nabais, C., Carvalho, A., Vieira, J., 2016. tracheideR – an R package to standardize tracheidograms. Dendrochronologia 37, 64–68.

Castagneri, D., Petit, G., Carrer, M., 2015. Divergent climate response on hydraulic-related xylem anatomical traits of Picea abies along a 900-m altitudinal gradient. Tree Physiol. 35, 1378–1387.

Chun, K.P., Mamet, S.D., Metsaranta, J., Barr, A., Johnstone, J., Wheater, H., 2017. A novel stochastic method for reconstructing daily precipitation times-series using tree-ring data from the western Canadian Boreal Forest. Dendrochronologia 44, 9–18.

Foresee, F.D., Hagan, M.T., 1997. Gauss-Newton approximation to Bayesian Learning. Proceedings of the 1997 International Joint Conference on Neural Networks.

Fritts, H.C., 1976. Tree Rings and Climate. Academic Press, London, pp. 567.

Gričar, J., 2010. Xylem and phloem formation in sessile oak from Slovenia in 2007. Wood Research 55, 15–22.

Jevšenak, J., Levanič, T., 2015. Dendrochronological and wood-anatomical features of differently vital pedunculate oak (*Quercus robur* L.) stands and their response to climate. Topola 195/196, 85–96.

Jevšenak, J., Levanič, T., 2016. Should artificial neural networks replace linear models in tree ring based climate reconstructions? Dendrochronologia 40, 102–109.

Jevšenak, J., Levanič, T., 2018. dendroTools: Nonlinear Methods for Analyzing Dendroclimatological Data. R Package Version 0.0.5. https://CRAN.R-project.org/package=dendroTools.

Kelley, D., Richards, C., 2017. Oce: Analysis of Oceanographic Data. R Package Version 0.9-21. https://CRAN.R-project.org/package=oce.

Lara, W., Bravo, F., Sierra, C.A., 2015. measuRing: an R package to measure tree-ring widths from scanned images. Dendrochronologia 34, 43–50.

Li, Z., Zhang, J., 2010. Dcv: Conventional Cross-Validation Statistics for Climate-Growth Model. R package version 0.1.1. https://CRAN.R-project.org/package=dcv.

Liang, W., Heinrich, I., Simard, S., Helle, G., Liñán, I.D., Heinken, T., 2013. Climate signals derived from cell anatomy of Scots pine in NE Germany. Tree Physiol. 33, 833–844.

Mackay, D.J.C., 1992. Bayesian Interpolation. Neural Comput. 4, 415–447.

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. Comput. Geosci. 37, 1234–1241.

Nguyen, D., Widrow, B., 1990. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. Proceedings of the IJCNN 3. IEEE 21–26.

Pérez-Rodríguez, P., Gianola, D., 2016. Brnn: Brnn (Bayesian Regularization for Feed-forward Neural Networks). R Package Version 0.6. http://CRAN.R-project.org/package=brnn.

Pérez-Rodríguez, P., Gianola, D., Weigel, K.A., Rosa, G.J.M., Crossa, J., 2013. Technical Note: an R package for fitting Bayesian regularized neural networks with applications in animal breeding. J. Anim. Sci. 91, 3522–3531.

R Core Team, 2017. A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. http://www.R-project.org/.

Rathgeber, C.B.K., Longuetaud, F., Mothe, F., Cuny, H., Le Moguedec, G., 2011. Phenology of wood formation: data processing, analysis and visualisation using R (package CAVIAR). Dendrochronologia 29, 139–149.

Rossi, S., Deslauriers, A., Anfodillo, T., Carraro, V., 2007. Evidence of threshold temperatures for xylogenesis in conifers at high altitudes. Oecologia 152, 1–12.

Schweingruber, F.H., 1981. Vrsic Krajanska Gora – PCAB – ITRDB YUGO001. https://www.ncdc.noaa.gov/paleo/study/4728.

Swidrak, I., Gruber, A., Oberhuber, W., 2011. Cambial activity and xylem cell development in Pinus cembra and Pinus sylvestris at their climatic limits in the Eastern Alps in 2007. Phyton-Annales Rei Botanicae 51, 299–313.

Touchan, R., Shishov, V.V., Meko, D.M., Nouiri, I., Grachev, A., 2012. Process based model sheds light on climate sensitivity of Mediterranean tree-ring width. Biogeosciences 9, 965–972.

van der Maaten, E., van der Maaten-Theunissen, M., Smiljanic, M., Rossi, S., Simard, S., Wilmking, M., Deslauriers, A., Fonti, P., von Arx, G., Bouriaud, O., 2016. DendrometeR: analyzing the pulse of trees in R. Dendrochronologia 40, 12–16.

van der Maaten-Theunissen, M., van der Maaten, E., Bouriaud, O., 2015. PointRes: an R package to analyze pointer years and components of resilience. Dendrochronologia 35, 34–38.

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

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

Wickham, H., 2016. Scales: Scale Functions for Visualization. R Package Version 0.4.1. https://CRAN.R-project.org/package=scales.

Wickham, H., Francois, R., Henry, L., Müller, K., 2017. Dplyr: A Grammar of Data Manipulation. R Package Version 0.7.3. https://CRAN.R-project.org/package=dplyr.

Yan, Y., 2016. MLmetrics: Machine Learning Evaluation Metrics. R Package Version 1.1.1. https://CRAN.R-project.org/package=MLmetrics.

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