Package 'dplRCon'

February 26, 2015

Type Package

Title Concordance for Dendroclimatology

Version 1.0

Date 2015-01-16

Author Maryann Pirie

Maintainer M R Pirie `<maryann.pirie@agresearch.co.nz>`

Description The concordance method is a non-parametric method based on bootstrapping that is used to test the hypothesis that two subsets of time series are similar in terms of mean, variance or both. This method was developed to address a concern within dendroclimatology that young trees may produce a differing climate response to older more established trees. Details of this method are available in Pirie, M. (2013). The Climate of New Zealand reconstructed from kauri tree rings: Enhancement through the use of novel statistical methodology. PhD. Dissertation, School of Environment and Department of Statistics, University of Auckland, New Zealand. This package also produces a figure with 3 panels, each panel is for a different climate variable. An example of this figure in included in ``On the influence of tree size on the climate - growth relationship of New Zealand kauri (Agathis australis): insights from annual, monthly and daily growth patterns. J Wunder, AM Fowler, ER Cook, M Pirie, SPJ McCloskey. Trees 27 (4), 937-948''. For further R functions for loading your own dendroclimatology datasets and performing dendrochronology analysis refer to the R package ``dplR: Dendrochronology Program Library in R''. The concordance procedure is intended to add to the standard dendrochronology techniques provided in ``dplR''.

License GPL

LazyLoad yes

LazyData yes

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-02-26 01:56:47

R topics documented:

`boot.0.20` *Bootstrapped matrix of time series for young material.*

Description

This is the bootstrapped matrices of mean time series produced from the "series.bootstrap" function.

Usage

```
data(boot.0.20)
```

Format

The format is: num [1:515, 1:999] NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...

Examples

```
data(boot.0.20)
## maybe str(boot.0.20) ; plot(boot.0.20) ...
## Not run:
# This dataset was produced by running the following.
data(ring.raw)
data(ring.stand)
data(dbh.po.nc)

#Subset "near-pith" is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))

boot.0.20   <-  series.bootstrap( spline200.sub0.20.n$sub.series.stand, stat, 999,
names.stat, aver.by.tree = FALSE)

boot.0.20 <- boot.0.20$boot.series.mean

## End(Not run)
```

`boot.20.2000` *Bootstrapped matrix of time series for old material.*

Description

This is the bootstrapped matrices of mean time series produced from the "series.bootstrap" function.

Usage

```
data(boot.20.2000)
```

Format

The format is: num [1:515, 1:999] 1.26 1.045 0.731 0.74 0.705 ...

Examples

```
data(boot.20.2000)
## maybe str(boot.20.2000) ; plot(boot.20.2000) ...

## Not run:
# This dataset was produced by running the following.
data(ring.raw)
data(ring.stand)
data(dbh.po.nc)

# Subset "far-pith" is the material further than 20cm from the estimated pith
spline200.sub20.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(200,200000))

boot.20.2000   <- series.bootstrap(spline200.sub20.2000.n$sub.series.stand, stat, 999,
```

```
names.stat, aver.by.tree = FALSE)

boot.20.2000 <- boot.20.2000$boot.series.mean

## End(Not run)
```

`climate.anom.dendro.season`

Seasonal Data

Description

Changes monthly climate data to seasonal climate data.

Usage

```
climate.anom.dendro.season(climate.data, year1, year2, year.start, is.anomaly)
```

Arguments

`climate.data`	This is a matrix, normally loaded into R in CSV form with the columns: year, jan, feb, mar,..,dec
`year1`	is the start of the period used to calculate climate anomalies.
`year2`	is the end of the period used to calculate climate anomalies.
`year.start`	is the start date of the climate data,
`is.anomaly`	accepts true or false. True will calculate the anomalies; departures from the mean from the period specified in year1 and year2

Details

This is required to change the monthly data, such as the data used in this package, to the seasons; Sept-Oct-Nov, Dec-Jan-Feb, Mar-Apr-May, Jun-Jul-Aug.

Value

Return a matrix of time series with col names season for the seasons including lagged seasons; "SON_2","DJF_2","MAM_2","JJA_2","SON_1","DJF_1","MAM_1", "JJA_1","SON","DJF","MAM","JJA";

Author(s)

Maryann Pirie

Examples

```
data(SOI)  # this is the Southern Oscillation Index data loaded with this package.
data(temperature) # this is the temperature data loaded with this package.
data(precipitation) # this is the precipitation data loaded with this package.
SOI.anom.season.data  <- climate.anom.dendro.season( SOI, 1933, 1992, 1876,
    is.anomaly="TRUE")
temp.anom.season.data <- climate.anom.dendro.season( temperature, 1933, 1992, 1876,
   is.anomaly="TRUE")
prec.anom.season.data <- climate.anom.dendro.season( precipitation, 1933, 1992, 1876,
   is.anomaly="TRUE")
```

`concordance.indices` *Concordance*

Description

This is an internal function called in the function "Overall Concordance". It is used to calculate the concordance indices for each time, t, using the bootstrapped means.

Usage

```
concordance.indices(sort.boot.x, sort.boot.y, t, trim.alpha, nx, ny)
```

Arguments

`sort.boot.x`	A matrix of the bootstrapped means from subset X
`sort.boot.y`	A matrix of the bootstrapped means from subset Y
`t`	The time index
`trim.alpha`	The level of outside trimming
`nx`	A vector providing the number of series at each time for subset X
`ny`	A vector providing the number of series at each time for subset Y

Details

The procedure for calculating the concordance is provided in "Concordance: A measure of similarity between matrices of time series with applications in dendroclimatology".

Value

A vector containing the a, b and the concordance

`correlation.function` *Performs correlations functions*

Description

Uses the imported climate variables and tree ring data to produce seasonal correlation functions. Also, uses the bootstrapped chronologies to produce confidence intervals.

Usage

```
correlation.function(climate.anom.season.data, site.chron.data, site.boot.data,
  period.RF, col.names.season, Climate.name, Subset.name)
```

Arguments

`climate.anom.season.data`	climate data anomalies for seasons
`site.chron.data`	site chronologies, matrix: rows=year, col=subset.1, subset.2,fullest
`site.boot.data`	bootstrapped site chronologies, list: matrices for each subset with: row=year, col=bootstrapped series
`period.RF`	the period used to calculate response functions, vector: (start,end)
`col.names.season`	col.names.season<- list("SON_2", "DJF_2", "MAM_2", "JJA_2", "SON_1", "DJF_1", "MAM_1", "JJA_1", "SON", "DJF", "MAM", "JJA")
`Climate.name`	name of the climate variable for which correlation functions are being calculated
`Subset.name`	names given to each of the subsets.

Value

`corr.site.1`	The correlations between the climate variable and the site chronology for the 1st subset.
`corr.site.2`	The correlations between the climate variable and the site chronology for the 12st subset.
`percentile.ci.1`	The percentile confidence intervals for the 1st subset.
`percentile.ci.2`	The percentile confidence intervals for the 2st subset.

Other summary varibles

`summary.ci.1`	
`summary.ci.2`	
`t.mean`	Test for correlation equal zero
`t.meanequal`	Test for correlations from the two subsets are equal
`percentile.ci`	

Note

site.chron.data, and site.boot.data must be in the same order and confidence intervals plotted are for the 1st two subsets.

Examples

```
## Not run:
period.RF<-c(1900,1990)
col.names.season <- list("SON_2", "DJF_2", "MAM_2", "JJA_2", "SON_1", "DJF_1", "MAM_1","JJA_1",
         "SON", "DJF", "MAM", "JJA")
##  Full dataset
site.full <- site.chron(spline200.sub0.2000.n$sub.series.stand, aver.by.tree=F)
site.chron.data <- cbind(site.full$aver.site, site.full$aver.site)
site.boot.full <- ts(boot.full$boot.series.mean, start=tsp(site.full$aver.site)[1] )
site.boot.data<-list(site.boot.full, site.boot.full)

corr.SOI.full<-correlation.function(SOI.anom.season.data, site.chron.data,site.boot.data,
      period.RF, col.names.season,
      Climate.name="SOI", Subset.name=c("0-20cm","20-200cm" ) )
corr.prec.full<-correlation.function(prec.anom.season.data, site.chron.data,site.boot.data,
     period.RF, col.names.season,
     Climate.name="SOI", Subset.name=c("0-20cm","20-200cm" ) )
corr.temp.full<-correlation.function(temp.anom.season.data, site.chron.data,site.boot.data,
     period.RF, col.names.season,
     Climate.name="SOI", Subset.name=c("0-20cm","20-200cm" ) )

## Near vs Far
site.0.20  <- site.chron(spline200.sub0.20.n$sub.series.stand, aver.by.tree=F)
site.20.200 <- site.chron(spline200.sub20.2000.n$sub.series.stand, aver.by.tree=F)
site.chron.data <- cbind(site.0.20$aver.site, site.20.200$aver.site)

site.boot.0.20 <- ts(boot.0.20$boot.series.mean, start=tsp(site.0.20$aver.site)[1] )
site.boot.20.200 <- ts(boot.20.2000$boot.series.mean, start=tsp(site.20.200$aver.site)[1] )
site.boot.data<-list(site.boot.0.20, site.boot.20.200)

corr.SOI<-correlation.function(SOI.anom.season.data, site.chron.data, site.boot.data,
   period.RF, col.names.season,
   Climate.name="SOI",Subset.name=c("0-20cm","20-200cm" ) )
corr.prec<-correlation.function(prec.anom.season.data, site.chron.data,        site.boot.data,
   period.RF, col.names.season,
   Climate.name="SOI", Subset.name=c("0-20cm","20-200cm" ) )
corr.temp<-correlation.function(temp.anom.season.data, site.chron.data, site.boot.data,
   period.RF, col.names.season,
   Climate.name="SOI", Subset.name=c("0-20cm","20-200cm" ) )

## End(Not run)
```

`dbh.po.nc` *Pith offset information*

Description

Includes: DBH, pithoffset estimation plus the method used in the estimation, the length of the outer non-crossdated distance and the length of the bark.

Usage

```
data(dbh.po.nc)
```

Format

A data frame with 135 observations on the following 6 variables.

`Series.ID` a factor with levels `HUP004A` `HUP004B` `HUP004C` `HUP004D` `HUP013A` `HUP013B` `HUP036A` `HUP036B` `HUP036C` `HUP036D` `HUP036E` `HUP037A` `HUP037B` `HUP037C` `HUP037D` `HUP040A` `HUP040B` `HUP040C` `HUP040D` `HUP040E` `HUP040F` `HUP050A` `HUP050B` `HUP050C` `HUP050D` `HUP052A` `HUP052B` `HUP052C` `HUP052D` `HUP055A` `HUP055B` `HUP055C` `HUP056A` `HUP056B` `HUP060A` `HUP060B` `HUP062A` `HUP062B` `HUP063A` `HUP063B` `HUP065A` `HUP065B` `HUP065C` `HUP066A` `HUP066B` `HUP066C` `HUP073A` `HUP073B` `HUP074A` `HUP074B` `HUP074C` `HUP079A` `HUP079B` `HUP079C` `HUP098A` `HUP098B` `HUP098C` `HUP100A` `HUP100B` `HUP100C` `HUP113A` `HUP113B` `HUP113C` `HUP125A` `HUP125B` `HUP125C` `HUP137A` `HUP137B` `HUP141A` `HUP141B` `HUP141C` `HUP141D` `HUP142A` `HUP142B` `HUP142C` `HUP142D` `HUP163A` `HUP163B` `HUP163C` `HUP163D` `HUP163E` `HUP189A` `HUP189B` `HUP189C` `HUP189D` `HUP197A` `HUP197B` `HUP197C` `HUP197D` `HUP197E` `HUP197F` `HUP198A` `HUP198B` `HUP198C` `HUP198D` `HUP198E` `HUP199A` `HUP199B` `HUP199C` `HUP199D` `HUP199E` `HUP199F` `HUP199g` `HUP200A` `HUP200B` `HUP200C` `HUP200D` `HUP200E` `HUP200F` `HUP201A` `HUP201B` `HUP201C` `HUP201D` `HUP202A` `HUP202B` `HUP202C` `HUP202D` `HUP202E` `HUP202F` `HUP208A` `HUP208B` `HUP208C` `HUP228A` `HUP228B` `HUP228C` `HUP228D` `HUP228E` `HUP232A` `HUP232B` `HUP232C` `HUP242A` `HUP242B` `HUP242C` `HUP242D` `HUP242E` `HUP247A` `HUP247B` `HUP247C` `HUP247D` `HUP248A` `HUP248B` `HUP248C` `HUP249A` `HUP249B` `HUP251A` `HUP251B` `HUP251C` `HUP251D` `HUP251E` `HUP251F` `HUP251g` `HUP251h` `HUP251I` `HUP257A` `HUP257B` `HUP257C` `HUP257D` `HUP259A` `HUP259B` `HUP259C` `HUP259D` `HUP279A` `HUP279B` `HUP301A` `HUP301B` `HUP301C` `HUP301D` `HUP303A` `HUP303B` `HUP303C` `HUP304A` `HUP304B` `HUP306A` `HUP306B` `HUP306C` `HUP310A` `HUP310B` `HUP310C` `HUP400A` `HUP400B`

`DCH` a numeric vector

`pithoffset` a numeric vector

`method` a factor with levels `BE_DBH_1M` `BE_length_min_tree` `Duncan` `partial_arc` `pith` `RADIUS`

`Outer.dis.non.crossdated` a numeric vector

`Bark` a numeric vector

Source

With permission from School of Environment, University of Auckland

Examples

```
data(dbh.po.nc)
str(dbh.po.nc)
```

`exact.ci` *Exact Confidence Intervals*

Description

Calculate the exact confidence intervals for proportions, based in Pearson's approach.

Usage

```
exact.ci(p.adjust, size.x, size.y, i)
```

Arguments

`p.adjust`	A vector of concordance indices, these are proportions for which you want to calculate the confidence interval.
`size.x`	A vector of the number of series used to calculate each index in matrix X.
`size.y`	A vector of the number of series used to calculate each index in matrix Y.
`i`	A recursive variables (i.e. use this function in Apply.)

Value

A vector with 2 elements indicating the lower and upper confidence intervals for the concordance indices.

`Example 1` *Example 1: Producing correlation figure*

Description

This describes the workflow required to generate the correlation figure published in "On the influence of tree size on the climate - growth relationship of New Zealand kauri (Agathis australis): insights from annual, monthly and daily growth patterns. J Wunder, AM Fowler, ER Cook, M Pirie, SPJ McCloskey. Trees 27 (4), 937-948".

Examples

```
 ## Not run:
#library('dplRCon')
#loading data
data(ring.raw)
data(ring.stand)
data(dbh.po.nc)
data(SOI)
data(temperature)
data(precipitation)
```

```
#Subset "near-pith" is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n   <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))
# Subset "far-pith" is the material further than 20cm from the estimated pith
spline200.sub20.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(200,200000))
# Whole dataset, through truncated functions to get in the same format as the above two datasets
spline200.sub0.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(00,200000))

#series.bootstapped
boot.0.20   <-  series.bootstrap( spline200.sub0.20.n$sub.series.stand, stat, 999,
names.stat, aver.by.tree = FALSE)
boot.20.2000   <- series.bootstrap(spline200.sub20.2000.n$sub.series.stand, stat, 999,
names.stat, aver.by.tree = FALSE)
boot.full   <- series.bootstrap(spline200.sub0.2000.n$sub.series.stand, stat, 999,
names.stat, aver.by.tree = FALSE)

#calculate seasonal climate
SOI.anom.season.data  <- climate.anom.dendro.season( SOI, 1933, 1992, 1876, is.anomaly="TRUE")
temp.anom.season.data <- climate.anom.dendro.season( temperature, 1933, 1992, 1876,
                is.anomaly="TRUE")
prec.anom.season.data <- climate.anom.dendro.season( precipitation, 1933, 1992, 1876,
                is.anomaly="TRUE")

# Calculate the correlations for each of the climate variables
period.RF<-c(1900,1990)
col.names.season <- list("SON_2", "DJF_2", "MAM_2", "JJA_2", "SON_1", "DJF_1", "MAM_1","JJA_1",
"SON", "DJF", "MAM", "JJA")

## Full dataset
site.full <- site.chron(spline200.sub0.2000.n$sub.series.stand, aver.by.tree=FALSE)
site.chron.data <- cbind(site.full$aver.site, site.full$aver.site)
site.boot.full <- ts(boot.full$boot.series.mean, start=tsp(site.full$aver.site)[1] )
site.boot.data<-list(site.boot.full, site.boot.full)
corr.SOI.full<-correlation.function(SOI.anom.season.data, site.chron.data, site.boot.data,
      period.RF,col.names.season, Climate.name="SOI",
      Subset.name=c("0-20cm","20-200cm" ) )
corr.prec.full<-correlation.function(prec.anom.season.data, site.chron.data, site.boot.data,
      period.RF, col.names.season, Climate.name="SOI",
      Subset.name=c("0-20cm","20-200cm" ) )
corr.temp.full<-correlation.function(temp.anom.season.data, site.chron.data, site.boot.data,
      period.RF, col.names.season, Climate.name="SOI",
      Subset.name=c("0-20cm","20-200cm" ) )


## Near vs Far
site.0.20  <- site.chron(spline200.sub0.20.n$sub.series.stand, aver.by.tree=FALSE)
site.20.200 <- site.chron(spline200.sub20.2000.n$sub.series.stand, aver.by.tree=FALSE)
site.chron.data <- cbind(site.0.20$aver.site, site.20.200$aver.site)

site.boot.0.20 <- ts(boot.0.20$boot.series.mean, start=tsp(site.0.20$aver.site)[1] )
site.boot.20.200 <- ts(boot.20.2000$boot.series.mean, start=tsp(site.20.200$aver.site)[1] )
site.boot.data<-list(site.boot.0.20, site.boot.20.200)
```

```
corr.SOI<-correlation.function(SOI.anom.season.data, site.chron.data,site.boot.data, period.RF,
col.names.season, Climate.name="SOI", Subset.name=c("0-20cm","20-200cm" ) )
corr.prec<-correlation.function(prec.anom.season.data, site.chron.data, site.boot.data, period.RF,
col.names.season, Climate.name="SOI",  Subset.name=c("0-20cm","20-200cm" ) )
corr.temp<-correlation.function(temp.anom.season.data, site.chron.data, site.boot.data, period.RF,
col.names.season, Climate.name="SOI",Subset.name=c("0-20cm","20-200cm" ) )

#Plotting
Figure.correlation.barplot(corr.temp, corr.prec, corr.SOI,
        corr.temp.full, corr.prec.full, corr.SOI.full,
        col.names.season)

## End(Not run)
```

`Example 2` *Example 2: Calculate the cordance*

Description

This discribes the workflow required to calculate the concordance published in the Application section of the "Concordance: A measure of similarity between matrices of time series with applications in dendroclimatology".

Examples

```
#library('dplRCon')
#loading data
data(ring.raw)
data(ring.stand)
data(dbh.po.nc)

#Subset "near-pith" is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n   <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))
# Subset "far-pith" is the material further than 20cm from the estimated pith
spline200.sub20.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(200,200000))
# Whole dataset, through truncated functions to get in the same formate as the above two datasets
spline200.sub0.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(00,200000))

## Not run:
#series.bootstapped
boot.0.20   <-  series.bootstrap( spline200.sub0.20.n$sub.series.stand, stat, 999,
names.stat, aver.by.tree = FALSE)
boot.20.2000   <- series.bootstrap(spline200.sub20.2000.n$sub.series.stand, stat, 999,
names.stat, aver.by.tree = FALSE)

boot.0.20 <- boot.0.20$boot.series.mean
boot.20.2000 <- boot.20.2000$boot.series.mean

## End(Not run)
```

```
data(boot.0.20)
data(boot.20.2000)

overall.precision.HUP <- overall.concordance.period(spline200.sub20.2000.n$sub.series.stand ,
spline200.sub0.20.n$sub.series.stand, boot.20.2000,
boot.0.20 ,1 , concordance.indices,
c(1880,1999), trim.alpha=0.005, concordance.beta=0.5)


figure.function.concordance (overall.precision.HUP, x.lim=c(1880,2000))
```

`Figure.correlation.barplot` *Produces a Figure of the correlation coefficients.*

Description

Produce a figure with 3 panels, each panel is for a different climate variable. An example of this figure in included in "On the influence of tree size on the climate - growth relationship of New Zealand kauri (Agathis australis): insights from annual, monthly and daily growth patterns. J Wunder, AM Fowler, ER Cook, M Pirie, SPJ McCloskey. Trees 27 (4), 937-948"

Usage

```
Figure.correlation.barplot(corr.1, corr.2, corr.3, corr.1.full, corr.2.full,
  corr.3.full, col.names.season)
```

Arguments

`corr.1`	the output from using function 'correlation.function' for the 1st climate variable comparing near and far-pith. Depending on the order of the inputs into the function 'correlation.function' will determine the color of the resultant box-plot (1st item is black, and 2nd item is gray)
`corr.2`	same as corr.1 but for the second climate variable
`corr.3`	same as corr.1 but for the third climate variable
`corr.1.full`	the output from using function 'correlation.function' for the full dataset. correlations for the full dataset are shown as a grey dashed line.
`corr.2.full`	same as corr.1.full but for the second climate variable
`corr.3.full`	same as corr.1.full but for the third climate variable
`col.names.season`	col.names.season<- list("SON_2", "DJF_2", "MAM_2", "JJA_2", "SON_1", "DJF_1", "MAM_1", "JJA_1", "SON", "DJF", "MAM", "JJA")

Value

This returns a figure.

Examples

```
## Not run:  Figure.correlation.barplot(corr.temp, corr.prec, corr.SOI,
   corr.temp.full, corr.prec.full, corr.SOI.full, col.names.season)
## End(Not run)
```

`figure.function.concordance`

Figure: Plots the Concordances

Description

Plots the concordance indices. This is used to generate the Figure in Concordance: A measure of similarity between matrices of time series with applications in dendroclimatology".

Usage

```
figure.function.concordance(con.indices, x.lim)
```

Arguments

`con.indices`	This is the output generated from the function overall.concordance.period.
`x.lim`	This is a vector with the start and end year for the required plot.

Value

A figure with the concordance indices and their confidence intervals, a smoother fitted through the indices and the confidence interval for the mean concordance.

Author(s)

Maryann Pirie

Examples

```
## Not run:
#Subset "near-pith" is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))
spline200.sub20.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(200,200000))
boot.0.20   <-  series.bootstrap( spline200.sub0.20.n$sub.series.stand, stat, 999,
    names.stat, aver.by.tree = FALSE)
boot.20.2000   <- series.bootstrap(spline200.sub20.2000.n$sub.series.stand, stat, 999,
   names.stat, aver.by.tree = FALSE)

overall.precision.HUP <- overall.concordance.period(spline200.sub20.2000.n$sub.series.stand ,
   spline200.sub0.20.n$sub.series.stand,
   boot.20.2000$boot.series.mean,  boot.0.20$boot.series.mean ,1 , concordance.indices,
   c(1880,1999), trim.alpha=0.005, concordance.beta=0.5)

figure.function.concordance (overall.precision.HUP, x.lim=c(1880,2000))
## End(Not run)
```

`function.id` *Core IDs*

Description

Take the series ID's and convert into a useable format. This produces series ID in a form that can be compared to pith offset file.

Usage

```
function.id(cores, site.id, tree.id, core.id)
```

Arguments

`cores`	A vector containing core Ids.
`site.id`	A vector specifying the position of start and end for the site ID.
`tree.id`	A vector specifying the position of start and end for the tree ID.
`core.id`	A vector specifying the position of start and end for the core ID.

Value

A vector of core ids, re-formatted.

`function.id.master` *Master ID*

Description

Produces series ID in a form that can be compared to pith offset file for the master chronology.

Usage

```
function.id.master(cores, site.id, tree.id, core.id)
```

Arguments

`cores`	A vector containing core Ids
`site.id`	A vector specifying the position of start and end for the site ID.
`tree.id`	A vector specifying the position of start and end for the tree ID.
`core.id`	A vector specifying the position of start and end for the core ID.

Value

A vector of core ids, re-formatted.

`names.stat` *names.stat*

Description

names.stat

Usage

```
names.stat
```

Format

```
chr [1:2] "mean" "median"
```

`overall.concordance.period` *Overall Concordance period*

Description

Produces the concordance indices for each year over the period of interest and the average of these indices. Trimmed and different levels of concordance indices in the overall concordance, size adjusted (mean concordance indices)

Usage

```
overall.concordance.period(x, y, x.boot, y.boot, min.series,
  concordance.indices, period, trim.alpha = 0, concordance.beta = 0.5)
```

Arguments

`x`	A matrix containing the standardized ring width indices for subset X, column heading tree ID, row names is time
`y`	As x, but for subset Y
`x.boot`	the bootstrapped replicates for matrix x. with bootstrapped replicate as row names, and time as the column headings
`y.boot`	the bootstrapped replicates for matrix y.
`min.series`	is a number indicating the minimum number of series for calculating the concordance statistic, recommend 10.
`concordance.indices`	The function used for calculating the concordance indices for each t.
`period`	Time period for which the concordance is to be calculated, vector of form c(start, end)

`trim.alpha`	this is the amount of trimming of the extreme concordance indices, default is 0 (no trimming). Recommend that a 0.005 be used.
`concordance.beta`	This is used to calculated the proportion of indices above this cut off. Default is proportion above 0.5.

Value

`pre.in`	Concordance indices.
`min.series`	Minimum number of series used to calculate concordance.
`num.series.at.t.x`	Number of series in matrix X for each time, t.
`num.series.at.t.y`	Number of series in matrix Y for each time, t.
`pre.in.greater.ts.period`	Concordance indices, as time series, when minimum number of series was available.
`period`	The time period used to calculate the concordance.
`pre.overall.period`	The Concordance Statistic.

Examples

```
#loading data
## Not run:
data(ring.raw)
data(ring.stand)
data(dbh.po.nc)

#Subset near-pith is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))
# Subset far-pith is the material further than 20cm from the estimated pith
spline200.sub20.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(200,200000))

# series.bootstapped
boot.0.20   <-  series.bootstrap( spline200.sub0.20.n$sub.series.stand,
   stat, 999, names.stat, aver.by.tree = FALSE)
boot.20.2000   <- series.bootstrap(spline200.sub20.2000.n$sub.series.stand,
   stat, 999, names.stat, aver.by.tree = FALSE)

overall.precision.HUP <- overall.concordance.period(spline200.sub20.2000.n$sub.series.stand ,
   spline200.sub0.20.n$sub.series.stand,
   boot.20.2000$boot.series.mean,  boot.0.20$boot.series.mean ,
   1 , concordance.indices, c(1880,1999), trim.alpha=0.005, concordance.beta=0.5)

## End(Not run)
```

`precipitation` *Monthly Precipitation Data*

Description

Monthly Precipitation Data

Usage

```
data(precipitation)
```

Format

A data frame with 153 observations on the following 13 variables.

`YEAR`	a numeric vector
`Jan`	a numeric vector
`Feb`	a numeric vector
`Mar`	a numeric vector
`Apr`	a numeric vector
`May`	a numeric vector
`Jun`	a numeric vector
`Jul`	a numeric vector
`Aug`	a numeric vector
`Sep`	a numeric vector
`Oct`	a numeric vector
`Nov`	a numeric vector
`Dec`	a numeric vector

Source

NIWA (The National Institute of Water and Atmosphere Research, New Zealand)

Examples

```
data(precipitation)
## maybe str(precipitation) ; plot(precipitation) ...
```

`ring.raw` *Matrix of time series of raw ring widths.*

Description

Matrix of time series of raw ring widths, each series in aligned in columns, with the column heading matching the series ID.

Usage

```
data(ring.raw)
```

Format

The format is: mts [1:515, 1:135] NA NA NA NA NA NA NA NA NA NA ... - attr(*, "dimnames")=List of 2 ..$: NULL ..$: chr [1:135] "HUP004A" "HUP004B" "HUP004C" "HUP004D" ... - attr(*, "tsp")= num [1:3] 1483 1997 1 - attr(*, "class")= chr [1:3] "mts" "ts" "matrix"

Source

Sourced from The School of Environment, University of Auckland with permission

Examples

```
data(ring.raw)
## maybe str(ring.raw) ; plot(ring.raw) ...
```

`ring.stand` *Matrix of time series for the standardised tree ring indices.*

Description

Matrix of time series of standardised ring widths, each series in aligned in columns, with the column heading matching the series ID.

Usage

```
data(ring.stand)
```

Format

The format is: mts [1:515, 1:135] NA NA NA NA NA NA NA NA NA NA ... - attr(*, "dimnames")=List of 2 ..$: NULL ..$: chr [1:135] "HUP004A" "HUP004B" "HUP004C" "HUP004D" ... - attr(*, "tsp")= num [1:3] 1483 1997 1 - attr(*, "class")= chr [1:3] "mts" "ts" "matrix"

Source

Sourced from The School of Environment, University of Auckland with permission

Examples

```
data(ring.stand)
## maybe str(ring.stand) ; plot(ring.stand) ...
```

`series.bootstrap` *Series bootstrap*

Description

This function calculate the bootstrapped replicated series for calculating concordance

Usage

```
series.bootstrap(data.boot, stat, R, names.stat, aver.by.tree.input)
```

Arguments

`data.boot`	Data matrix containing standardized ring width indices from which the bootstrapped replicates are calculated.
`stat`	A function giving the statistics that are to be calculated
`R`	The number of bootstrapped replicates
`names.stat`	the names of the statistics contained in function "stat"
`aver.by.tree.input`	This is a True/False. If True then averages tree series, then average tree means. If False then average all series

Value

`org.stat`	The original input statistics
`boot.stat`	The statistics of the bootstrapped replicates, c('mean','median')
`bias`	The bias of the boot.stat
`std.error`	The standard error of boot.stat
`boot.series.mean`	The bootstrapped replicates mean series
`boot.series.std`	The bootstrapped replicates standard deviation series

Examples

```
## Not run: #Subset "near-pith" is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n   <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))
boot.0.20   <-  series.bootstrap( spline200.sub0.20.n$sub.series.stand, stat, 999,
   names.stat, aver.by.tree = FALSE)
## End(Not run)
```

`site.chron` *site.chron*

Description

This function calculate the site chronology for the data matrix

Usage

```
site.chron(data.m, aver.by.tree.input)
```

Arguments

`data.m`	A data matrix contains standardized ring width indices
`aver.by.tree.input`	If True then averages tree series, then average tree means. If False then average all series

Value

`aver.site`	The average for the site.
`std.site`	The standard deviation of the site.
`aver.tree`	A time series for average of each tree.
`number.trees`	The total number of trees used to produce the site chronology.
`core.per.tree`	The number of cores for each tree.

Examples

```
## Not run:
site.full <- site.chron(spline200.sub0.2000.n$sub.series.stand, aver.by.tree=FALSE)

## End(Not run)
```

`SOI` *SOI data*

Description

Southern Oscillation Index Data

Usage

```
data(SOI)
```

Format

The format is: chr "SOI"

Source

BOM (Australian Bureau of Meteorology)

Examples

```
data(SOI)
## maybe str(SOI) ; plot(SOI) ...
```

`stat` *Stat*

Description

To be used internally in series.bootstrapped this is the summary that is calculated during bootstrapping.

Usage

```
stat(data)
```

Arguments

`data`	data matrix (produced within series.bootstrapped)

Value

results required output for series.bootstrapped

`SumNotNa` *Sum non Na's*

Description

Count the number of non na values. This is an internal function used in an apply function.

Usage

```
SumNotNa(x)
```

Arguments

`x`	Vector to be counted containing Na's

Value

The sum of the non na's values.

`temperature` *Monthly Temperature Data*

Description

Monthly Temperature Data

Usage

```
data(temperature)
```

Format

A data frame with 154 observations on the following 13 variables.

`YEAR`	a numeric vector
`Jan`	a numeric vector
`Feb`	a numeric vector
`Mar`	a numeric vector
`Apr`	a numeric vector
`May`	a numeric vector
`Jun`	a numeric vector
`Jul`	a numeric vector
`Aug`	a numeric vector
`Sep`	a numeric vector
`Oct`	a numeric vector
`Nov`	a numeric vector
`Dec`	a numeric vector

Source

NIWA (The National Institute of Water and Atmosphere Research, New Zealand)

Examples

```
data(temperature)
## maybe str(temperature) ; plot(temperature) ...
```

`TruncSeriesPithoffset` *Truncate series by range pithoffsets*

Description

The following function truncate the data by a given range from the estimated pith.

Usage

```
TruncSeriesPithoffset(file.raw, file.stand, pithoffset, range)
```

Arguments

`file.raw`	data file containing the raw ring widths, in mm
`file.stand`	data file containing the standardized ring widths
`pithoffset`	data set containing the pith offsets for each core, in mm.
`range`	The distance from the pith use for truncation, given in mm from the core. e.g. range <- c(1,200) truncates values outside this range.

Value

`sub.series.raw`	A truncated series of raw ring widths (in miromilmeter).
`sub.series.stand`	A truncated series of standardized ring widths (in miromilmeter).

Examples

```
## Not run:
data(ring.raw)
data(ring.stand)
data(dbh.po.nc)
#Subset near-pith is the material within 0 -20cm from the estimated pith
spline200.sub0.20.n   <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(1,200))
# Subset far-pith is the material further than 20cm from the estimated pith
spline200.sub20.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(200,200000))
# Whole dataset, through truncated functions to get in the same formate as the above two datasets
spline200.sub0.2000.n  <- TruncSeriesPithoffset( ring.raw, ring.stand, dbh.po.nc, c(00,200000))
## End(Not run)
```

Index