Package 'RolWinMulCor'

May 22, 2020

Type Package

Title Subroutines to Estimate Rolling Window Multiple Correlation

Version 0.4.0

Author Josue M. Polanco-Martinez [aut, cph, cre]

Maintainer Josue M. Polanco-Martinez <josue.m.polanco@gmail.com>

Depends R (>= 3.5.0), stats, gtools, zoo, pracma, colorspace

Description Rolling Window Multiple Correlation ('RolWinMulCor') estimates the rolling (running) window correlation for the bi- and multi-variate cases between regular (sampled on identical time points) time series, with especial emphasis to environmental data (although this can be applied to other kinds of data sets). 'RolWinMulCor' is based on the concept of rolling or running window and is useful to evaluate the evolution of correlation through time and time-scales. 'RolWinMulCor' contains four functions: (1) the first two functions are focus on the bi-variate case, one of them produces a simple plot of correlation coefficients and p-values (<=0.05) for only one window-length (time-scale), and the other function produces a heat map for the statistically significant (p-values <=0.05) correlation coefficients taking into account all the possible window-lengths (which are determined by the number of elements of the time series under analysis) or for a band of window-lengths; (2) the second two functions are designed to analyse the multi-variate case and follow the bivariate case to display visually the results although these two methods are different. The four functions contained in 'RolWinMulCor' are highly flexible since this contains a great number of parameters to control the estimation of correlation and the features of the plot output, e.g. to remove the (linear) trend contained in the time series under analysis, to choose different p-value correction methods (which are used to address the multiple comparison problem) or to personalise the plot output (e.g. this can be displayed in the screen or can be saved as PNG, JPEG, EPS or PDF formats). The 'RolWinMulCor' package also provides examples with synthetic and real environmental time series to exemplify its use. Methods derived from H. Abdi. (2007) <https://personal.utdallas.edu/~herve/Abdi-MCC2007-pretty.pdf>, J. M. Polanco-Martinez (2019) <doi:10.1007/s11071-019-04974-y>, and R. Telford (2013) <https://quantpalaeo.wordpress.com/2013/01/04/running-correlations-running-into-problems/>.

License GPL (>= 2)

Repository CRAN

Encoding UTF-8

LazyData true

Date/Publication 2020-05-22 16:10:02 UTC

NeedsCompilation no

R topics documented:

`RolWinMulCor-package` *Estimate the Rolling Window Multiple Correlation*

Description

'RolWinMulCor' estimates the rolling (running) window correlation for the bi- and multi-variate cases between regular (sampled on identical time points) time series, with especial emphasis to environmental data (although this can be applied to other kinds of data sets). 'RolWinMulCor' is based on the concept of rolling or running window correlation and is useful to evaluate the evolution of correlation through time and time-scales. 'RolWinMulCor' contains four functions: (1) the first two are focused on the bi-variate case, one of them produces a simple plot of correlation coefficients and p-values (<=0.05) for only one window-length (time-scale), and the other function produces a heat map for the statistically significant (p-values <=0.05) correlation coefficients taking into account all the possible window-lengths (which are determined by the number of elements of the time series under analysis) or for a band of window-lengths; (2) the second two functions are designed to analyse the multi-variate case and follows the bi-variate case to display visually the results although these two approaches are methodologically different. The four functions contained in 'RolWinMulCor' are highly flexible since this contains a great number of parameters to control the estimation of correlation and the features of the plot output, e.g. to remove the linear trend contained in the time series under analysis, to choose different p-value correction methods (which are used to address the multiple comparison problem) or to personalise the plot output (e.g. this can be displayed in the screen or can be saved as PNG, JPG, EPS or PDF formats). The 'RolWinMulCor' package also provides examples with synthetic and real environmental time series to exemplify its use. We would like to highlight that, to the best of our knowledge, there are few R packages (probably the one) on CRAN that estimate rolling window correlation and produce a heat map for the bi-variate case and, especially, for the multi-variate case.

Details

Package:	RolWinMulCor
Type:	Package
Version:	0.4
Date:	2020-05-21
License:	GPL (>= 2)
LazyLoad:	yes

RolWinMulCor package contains four functions: (1) `rolwincor_1win` (estimates and plots the rolling window correlation for the bi-variate case for only one window-length or time-scale for the time series under study), (2) `rolwincor_heatmap` (estimates and plots as a heat map the statistically significant (p-values <=0.05) correlation coefficients taking into account all the possible window-lengths that are determined by the number of elements of the time series under or a band of window-lengths and plot the the correlation coefficients and their respective p-values as a heat map), (3) `rolwinmulcor_1win` (estimates and plots the rolling window correlation for the multi-variate case for only one window-length or time-scale for the time series under study) and, (4) `rolwinmulcor_heatmap` (estimates and plots the heat map for the multi-variate case). The bi-variate case follow from a methodological point of view to Telford (2013) and Polanco-Martínez (2019) whereas the multi-variate case follow to Abdi (2007).

Note

Dependencies: *stat*, *gtools*, *zoo*, *pracma* and *colorspace*.

Author(s)

Josué M. Polanco-Martínez (a.k.a. jomopo).
DeustoTech - Deusto Institute of Technology,
Faculty of Engineering, University of Deusto,
Avda. Universidades, 24, Bilbao, SPAIN.
Web1: https://scholar.google.es/citations?user=8djLIhcAAAAJ&hl=en.
Web2: http://www.researchgate.net/profile/Josue_Polanco-Martinez.
Email: <josue.m.polanco@gmail.com>, <josue.polanco@deusto.es>

Acknowledgement:
The author acknowledges to the SEPE (Spanish Public Service of Employment) for its funding support. Special thanks to the CRAN team (in particular to Martina Schmirl and Jelena Saf), and to Ana-Maria Hereş and Jorge Curie for their helpful comments on the package.

References

Abdi H. Multiple correlation coefficient, in Encyclopedia of Measurement and Statistics, N. J. Salkind, Ed. Sage, Thousand Oaks, CA, USA, 2007; 648-651.
<URL: https://personal.utdallas.edu/~herve/Abdi-MCC2007-pretty.pdf>.

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B, 57 (1),

289-300. <URL: https://rss.onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1995.tb02031.x>.

Polanco-Martínez, J. M. (2019). Dynamic relationship analysis between NAFTA stock markets using nonlinear, nonparametric, non-stationary methods. Nonlinear Dynamics, 97(1), 369-389. <URL: https://doi.org/10.1007/s11071-019-04974-y>.

Telford, R.: Running correlations – running into problems (2013). <URL: https://quantpalaeo.wordpress.com/2013/01/04/running-correlations-running-into-problems/>.

`rolwincor_1win` *Estimate and plot the Rolling Window Correlation for the bi-variate case*

Description

The `rolwincor_1win` function estimates the rolling (running) window correlation between TWO time series (bi-variate case) sampled on identical time points for ONLY ONE window-length and plots the correlation coefficients and their respective p-values. To carry out the computational implementation we follow to Telford (2013) and Polanco-Martínez (2019). The `rolwincor_1win` function is highly flexible since this contain a great number of parameters to control the estimation of correlation and the features of the plot output. For example, `rolwincor_1win` function contain parameters to remove the (linear) trend contained in the time series under analysis, to choose different p-value correction methods (which are used to address the multiple comparison problem), or to personalise the plot output. A list of parameters are described in the following lines.

Usage

```
rolwincor_1win(inputdata, varnametsX="", varnametsY="", units="",
               coltsX="black", coltsY="blue", colCOEF="black",
               colPVAL="gray", CEXLAB=1.15, CEXAXIS=1.05, LWDtsX=1,
               LWDtsY=1, LWDcoef=1, LWDpval=1, CorMethod="spearman",
               widthwin=3, Align="center", pvalcorectmethod="BH",
               rmltrd="Y", Scale="Y", device="screen", Hfig=1200,
               Wfig=900, resfig=150, Hpdf=5.15, Wpdf=13.5,
               HeigWin1=2.05, HeigWin2=2.75, ofilename)
```

Arguments

`inputdata`	Matrix of 3 columns: time, first variable (e.g. X), and second variable (e.g. Y).
`varnametsX, varnametsY`	Names of the first (e.g. X) and second (e.g. Y) variable. Please note that the names of these variables MUST be defined.
`units`	Time's unit (e.g. days, weeks, years, etc.) for the variables under analysis. Please note that the units MUST be defined.

`coltsX, coltsY` The colors to be used when the first and second variables are plotted, by default the colors are "black" and "blue," respectively, but other colors can be used.

`colCOEF, colPVAL`
The colors to be used when the correlation coefficients and their corresponding p-values are plotted, by default the colors are "black" and "gray," but other colors can be used.

`CEXLAB, CEXAXIS`
These parameters are used to plot the sizes of the X-axis and Y-axis labels and X- and Y-axis, by default these parameters have values of 1.15 and 1.05, respectively, but it is possible to use other values.

`LWDtsX, LWDtsY` Line-widths for the first and the second variable when these are plotted, by default these have values of 1, but other values (widths) can be used.

`LWDcoef, LWDpval`
The line-widths to be used when the correlation coefficients and their respective p-values are plotted, by default these parameters have a value of 1, but it is possible to use other values.

`CorMethod` The method used to estimate the correlations, by default is "spearman," but other options ("pearson" and "kendall") are available (please look at: R>?cor.test).

`widthwin` Window's size to compute the rolling window correlations, this MUST be odd (i.e. of the form **2m + 1**, where **m** is a natural number), the minimum value is 3 (the default value), but this information MUST be provided.

`Align` To align the rolling object, RolWinMulCor ONLY uses the "center" option (please look at: R>?running) to ensure that variations in the correlation are aligned with the variations in the relationship of the time series under study rather than being shifted (Polanco-Martínez, 2019).

`pvalcorectmethod`
The p-value correction method to be used, by default the method of Benjamini and Hochberg (BH) (1995) is used since this is less conservative and performs much better than Bonferroni, but other six methods (e.g. Holm, Bonferroni, etc.) are available (please look at: R>?p.adjust).

`rmltrd` Remove (by default is "Y" or "y"; please use "N" or "n" otherwise) the linear trend in the two time series under analysis.

`Scale` Scale (by default is "Y" or "y"; please use "N" or "n" otherwise) is used to "normalize" or "standardize" the time series under analysis.

`device` Kind of plot output (please look at: R>?device), there are five options: "png", "jpeg/jpg", "eps", "pdf" and "screen." By default device is "screen."

`Hfig, Wfig` Plot's height and width (for the device) for "png" and "jpg" format (look at R>?png or R>?jpg), by default Hfig and Wfig have values of 1200 and 900, but it is possible to use other values.

`resfig` Image resolution (in "ppi") for the plot in "png" and "jpg" format (look at R>?png, R>?jpg, or R>?jpeg), by default this has a value of 150, but it is possible to use other image resolutions.

`Hpdf, Wpdf` Plot's height and width (for the device) for "pdf" or "eps" format (>R?pdf or >R?postscript), by default Hpdf and Wpdf have values of 5.15 and 13.5, but other values can be used.

`HeigWin1, HeigWin2`
: Proportion of window's size to plot the time series under analysis (HeigWin1) and the rolling window correlation coefficients and p-values (HeigWin2) (look at: R>?layout to get more information about "layout"). By default HeigWin1 and HeigWin2 have values of 2.05 and 2.75, but other values can be used.

`ofilename`
: Output file name.

Details

The `rolwincor_1win` function estimates the rolling window correlation between TWO time series (bi-variate case) sampled on identical time points for ONLY ONE window-length and plots the rolling correlation coefficients and their respective p-values. `rolwincor_1win` uses the functions *running* (package:gtools), the native R functions *cor*, *cor.test*, and *p.adjust* (package:stats), and some pieces of code written specifically to our R RolWinMulCor package.

Value

Outputs:

Plot output: *screen* or *'plot_bivariate_' + 'ofilename + .png, .jpg, .eps* or *.pdf'*.

Numerical output: a list containing *Correlation_coefficients*, *P_values_corrected*, and *P_values_not_corrected*, which are self-explanatory.

Author(s)

Josué M. Polanco-Martínez (a.k.a. jomopo).
DeustoTech - Deusto Institute of Technology,
Faculty of Engineering, University of Deusto,
Avda. Universidades, 24, Bilbao, SPAIN.
Web1: https://scholar.google.es/citations?user=8djLIhcAAAAJ&hl=en.
Web2: http://www.researchgate.net/profile/Josue_Polanco-Martinez.
Email: <josue.m.polanco@gmail.com>, <josue.polanco@deusto.es>

References

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B, 57 (1), 289-300. <URL: https://rss.onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1995.tb02031.x>.

Polanco-Martínez, J. M. (2019). Dynamic relationship analysis between NAFTA stock markets using nonlinear, nonparametric, non-stationary methods. Nonlinear Dynamics, 97(1), 369-389. <URL: https://doi.org/10.1007/s11071-019-04974-y>.

Telford, R.: Running correlations – running into problems (2013). <URL: https://quantpalaeo.wordpress.com/2013/01/04/running-correlations-running-into-problems/>.

Examples

```
# Loading packages
library("RolWinMulCor")
library("gtools")
library("pracma")
# Loading data set
data(synthetic_data)
# Testing the function 'rolwincor_1win' (bi-variate case)
# Window length = 21 and plot output in screen format
test1_rolwincor_1win <- rolwincor_1win(synthetic_data[,1:3], varnametsX="X",
                        varnametsY="Y", units="NU", widthwin=21,
                        device="screen")
# Window length = 51 and plot output in screen format
test2_rolwincor_1win <- rolwincor_1win(synthetic_data[,1:3], varnametsX="X",
                        varnametsY="Y", units="NU", widthwin=51,
                        device="screen")
# Window length = 51 and plot output in PDF format

 test3_rolwincor_1win <- rolwincor_1win(synthetic_data[,1:3], varnametsX="X",
                        varnametsY="Y", units="NU", widthwin=51,
                        device="pdf", ofilename="test3")
```

`rolwincor_heatmap` *Estimate the Rolling Window Correlation for the bi-variate case and plot the results as a heat map*

Description

The `rolwincor_heatmap` function estimates the rolling (running) window correlation between TWO time series (bi-variate case) sampled on identical time points for all the possible (i.e. from 3 to the number of elements of the time series under analysis) window-lengths or for a band of window-lengths and plots the correlation coefficients and their respective p-values (<= 0.05) as a heat map. To carry out the computational implementation we extend the works of Telford (2013) and Polanco-Martínez (2019). The `rolwincor_heatmap` function is highly flexible since this contains a great number of parameters to control the estimation of correlation and features of the plot output. A list of parameters are described in the following lines.

Usage

```
rolwincor_heatmap(inputdata, varnametsX="", varnametsY="", units="",
                  coltsX="black", coltsY="blue", CEXLAB=1.15, CEXAXIS=1.5,
                 LWDtsX=1, LWDtsY=1, CorMethod="spearman", typewidthwin="FULL",
                  widthwin_1=3, widthwin_N=dim(inputdata)[1],
                  Align="center", pvalcorectmethod="BH", rmltrd="Y",
                  Scale="Y", device="screen", Hfig=900, Wfig=900,
                  resfig=150, Hpdf=7, Wpdf=7, NUMLABX=5, ofilename)
```

Arguments

`inputdata`	Matrix of 3 columns: time, first variable (e.g. X), and second variable (e.g. Y).
`varnametsX, varnametsY`	Names of the first (e.g. X) and second (e.g. Y) variable. Please note that the names of these two variables MUST be defined.
`units`	Time's unit (e.g. days, weeks, years, etc.) for the variables under analysis. Please note that the units MUST be defined.
`coltsX, coltsY`	The colors to be used to plot the two variables under study, by default the colors are "black" and "blue," respectively, but it is possible to use other colors.
`CEXLAB, CEXAXIS`	These parameters are used to plot the sizes of X-axis and Y-axis labels and X- and Y-axis, by default these parameters have values of 1.15 and 1.5, respectively, but it is possible to use other values.
`LWDtsX, LWDtsY`	These parameters are used to define the line-widths when the variables are plotted, by default these have values of 1, but other values (widths) can be used.
`CorMethod`	The method used to estimate the correlations, by default is "spearman" but other options ("pearson" and "kendall") are available (please look at: R>?cor.test).
`typewidthwin`	There are two options to estimate the rolling correlations and to plot the heat map: (1) typewidthwin="PARTIAL" to plot a band of windows from widthwin_1 to widthwin_N (both parameters were previously defined), or (2) typewidthwin="FULL" to plot all the possible window-lengths (from 3 to dim(inputdata)[1]), by default is FULL.
`widthwin_1`	First value for the size (length) of the windows when the option typewidthwin="PARTIAL" is selected, the minimum value is 3 (the default value), but you should define this parameter (please note that widthwin_1 < widthwin_N).
`widthwin_N`	Last value for the size (length) of the windows when the option typewidthwin="PARTIAL" is selected, by default is dim(inputdata)[1], but you should define this parameter (please note that widthwin_1 < widthwin_N).
`Align`	To align the rolling object, RolWinMulCor ONLY uses the "center" option (please look at: R>?running) to ensure that variations in the correlation are aligned with the variations in the relationship of the time series under study rather than being shifted (Polanco-Martínez 2019).
`pvalcorectmethod`	The p-value correction method to be used, by default the method of Benjamini and Hochberg (BH) (1995) is used since this is less conservative and performs much better than Bonferroni, but other six methods (e.g. Holm, Bonferroni, etc.) are available (please look at: R>?p.adjust).
`rmltrd`	Remove (by default is "Y" or "y"; please use "N" or "n" otherwise) the linear trend in the time series under analysis.
`Scale`	Scale (by default is "Y" or "y"; please use "N" or "n" otherwise) is used to "normalize" or "standardize" the time series under analysis.
`device`	Kind of plot output (please look at: R>?device), there are five options: "png", "jpeg/jpg", "eps", "pdf" and "screen." By default device is "screen."

`Hfig, Wfig`	Plot's height and width (for the device) for "png" and "jpg" format (look at R>?png or R>?jpg), by default Hfig and Wfig have values of 900, but other values can be used.
`resfig`	Image resolution (in "ppi") for the plot in "png" and "jpg" format (look at R>?png , R>?jpg or R>?jpeg), by default this has a value of 150, but other image resolutions can be used.
`Hpdf, Wpdf`	Plot's height and width (for the device) for "pdf" or "eps" format (>R?pdf or >R?postscript), by default Hpdf and Wpdf are equal to 7, but other values can be used.
`NUMLABX`	Number of labels for (all) the X's axis, by the default is 5, but other number of labels can be utilized.
`ofilename`	Output file name.

Details

The `rolwincor_heatmap` function estimates the rolling window correlation between TWO time series (bi-variate case) sampled on identical time points for all the possible (i.e. from 3 to the number of elements of the time series under analysis) window-lengths or for a band of window-lengths and plots the rolling correlation coefficients and their respective p-values as a heat map. `rolwincor_heatmap` uses the functions *running* (package:gtools), the native R functions *cor*, *cor.test*, and *p.adjust* (package:stats), and some pieces of code written specifically to our R RolWinMulCor package.

Value

Outputs:

Plot output (heat map): *screen* or *'heatmap_bivariate_' + 'ofilename + .png, .jpg, .eps* or *.pdf'*.

Numerical output: two lists *matcor* and *pvalscor* containing the correlation matrix and their corresponding corrected p-values, *NoWindows* and *Windows* that contains the number of windows and the window-lengths.

Author(s)

Josué M. Polanco-Martínez (a.k.a. jomopo).
DeustoTech - Deusto Institute of Technology,
Faculty of Engineering, University of Deusto,
Avda. Universidades, 24, Bilbao, SPAIN.
Web1: https://scholar.google.es/citations?user=8djLIhcAAAAJ&hl=en.
Web2: http://www.researchgate.net/profile/Josue_Polanco-Martinez.
Email: <josue.m.polanco@gmail.com>, <josue.polanco@deusto.es>

References

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B, 57 (1), 289-300. <URL: https://rss.onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1995.

tb02031.x>.

Polanco-Martínez, J. M. (2019). Dynamic relationship analysis between NAFTA stock markets using nonlinear, nonparametric, non-stationary methods. Nonlinear Dynamics, 97(1), 369-389. <URL: https://doi.org/10.1007/s11071-019-04974-y>.

Telford, R.: Running correlations – running into problems (2013). <URL: https://quantpalaeo.wordpress.com/2013/01/04/running-correlations-running-into-problems/>.

Examples

```
# Loading packages
library("RolWinMulCor")
library("gtools")
library("pracma")
# Loading data set
data(synthetic_data)
# Testing the function rolwincor_heatmap
#  typewidthwin="PARTIAL," window lengths from 21 to 31 and plot output in screen format
test1_rolwincor_heatmap <- rolwincor_heatmap(synthetic_data[,1:3], varnametsX="X",
                           varnametsY="Y", units="NU", typewidthwin="PARTIAL",
                           widthwin_1=21, widthwin_N=31, device="screen",)
# This example could takes a long time!
# typewidthwin="FULL" and pot output in PDF format

test2_rolwincor_heatmap <- rolwincor_heatmap(synthetic_data[,1:3], varnametsX="X",
                           varnametsY="Y", units="NU", typewidthwin="FULL",
                           device="pdf", ofilename="test2")
```

---

rolwinmulcor_1win *Estimate and plot the Rolling Window Correlation for the multivariate case*

Description

The rolwinmulcor_1win function estimates the rolling (running) window correlation among multiple time series (multi-variate case) sampled on identical time points for ONLY ONE window-length and plots the correlation coefficients and their respective p-values. The multivariate case is based on the concept of multiple regression and generalizes the standard coefficient of correlation. We follow to Abdi (2007) to implement computationally this technique. The rolwinmulcor_1win function is highly flexible since this contains a great number of parameters to control the estimation of correlation and the features of the plot output. For example, rolwinmulcor_1win function contains parameters to remove the (linear) trend contained in the time series under analysis, to choose different p-value correction methods (which are used to address the multiple comparison problem), or to personalise the plot output. A list of parameters are described in the following lines.

Usage

```
rolwinmulcor_1win(inputdata, varnametsY="", varnametsX="", units="",
                  coltsY="black", coltsX="", LWDtsY=1, LWDtsX="",
                  LWDcoef=1, LWDpval=1, CEXLAB=1.15, CEXAXIS=1.05,
                  NUMLABX=5, parcen=c(0.5,25), colCOEF="black",
                  colPVAL="gray", rmltrd="Y", Scale="Y", widthwin=3,
                  Align="center", pvalcorectmethod="BH", device="screen",
                  Hfig=1200, Wfig=900, resfig=150, Hpdf=5.15, Wpdf=13.5,
                  HeigWin1=2.05, HeigWin2=2.75, ofilename)
```

Arguments

`inputdata`	Matrix of P columns: time, dependent variable (Y), and independent variables ($X_1, X_2, ..., X_{P-2}$).
`varnametsY`	Name of the dependent variable: Y. Please note that the name of this variable MUST be defined.
`varnametsX`	Name of the independent variables: $X_1, X_2, ..., X_{p-2}$. Please note that the names of these variables MUST defined in this way: varnametsX=paste("X1", "X2",..., sep=", ").
`units`	Time's unit (e.g. days, weeks, years, etc.) for the time series under analysis. Please note that the units MUST de defined.
`coltsY, coltsX`	The colors to be used when the dependent (Y) and independent ($X_1, X_2, ...$) variables are plotted, by default for the dependent variable the color is "black" (although other colors can be used) and the colors for the independent variables MUST be provided (e.g. coltsX=c("red","blue",...)).
`LWDtsY, LWDtsX`	These parameters are used to define the line-widths when the dependent and independent variables are plotted, by default and for the dependent variable this has a value of 1 (although other values (widths) can be used), and for the independent variables the line-widths MUST be provided (e.g. LWDtsX = c(1,2,...)).
`LWDcoef, LWDpval`	The line-widths to be used when the correlation coefficients and their respective p-values are plotted, by default these parameters have a value of 1, but other values (widths) can be used.
`CEXLAB, CEXAXIS`	The parameters are used to plot the sizes of the X-axis and Y-axis labels and X- and Y-axis, by default these parameters have values of 1.15 and 1.05, respectively, but it is possible to use other values.
`NUMLABX`	Number of labels for (all) the X's axis, by the default is 5, but it is possible to use other values.
`parcen`	These parameters contain two values: the first one is to control the position of the title, by default it is 0.5, but you should try with other close values to obtain the title centered, e.g. 0.4 or 0.8 (please avoid to use large values); the second value is to define the spaces between the names of variables, by default is 25 spaces, but you could try other values to fit properly the names of variables in the title. We use "mtext" to produce the title (please loot at R>?mtext for more information).

`colCOEF, colPVAL`
: The colors to be used when the correlation coefficients and their corresponding p-values are plotted, by default the colors are "black" and "gray," but it is possible to use other colors.

`rmltrd`
: Remove (by default is "Y" or "y"; please use "N" or "n" otherwise) the linear trend in the time series under analysis.

`Scale`
: Scale (by default is "Y" or "y"; please use "N" or "n" otherwise) is used to "normalize" or "standardize" the time series under analysis.

`widthwin`
: Window's size to compute the rolling window correlations, this MUST be odd (i.e. of the form **2m + 1**, where **m** is a natural number), the minimum values is 3 (the default value), but this information MUST be provided.

`Align`
: To align the rolling object, RolWinMulCor ONLY uses the "center" option (please look at: R>?rollapply) to ensure that variations in the correlation are aligned with the variations in the relationship of the time series under study rather than being shifted (Polanco-Martínez 2019).

`pvalcorectmethod`
: The p-value correction method to be used, by default the method of Benjamini and Hochberg (BH) (1995) is used since this is less conservative and performs much better than Bonferroni, but other six methods (e.g. Holm, Bonferroni, etc.) are available (please look at: R>?p.adjust).

`device`
: Kind of plot output (please look at: R>?device), there are five options: "png", "jpeg/jpg", "eps", "pdf" and "screen." By default device is "screen."

`Hfig, Wfig`
: Plot's height and width (for the device) for the plot in "png" and "jpg" format (look at R>?png or R>?jpg), by default Hfig and Wfig have values of 1200 and 900, but other values can be used.

`resfig`
: Image resolution (in "ppi") for the plot in "png" and "jpg" format (look at R>?png , R>?jpg or R>?jpeg), by default this has a value of 150, but other mage resolutions can be used.

`Hpdf, Wpdf`
: Plot's height and width (for the device) for "pdf" or "eps" format (>R?pdf or >R?postscript), by default Hpdf and Wpdf are equal to 5.15 and 13.5, but it is possible to use other values.

`HeigWin1, HeigWin2`
: Proportion of window's size to plot the time series under analysis (HeigWin1) and the rolling window correlation coefficients and p-values (HeigWin2) (look at: R>?layout to get more information). By default HeigWin1 and HeigWin2 have value of 2.05 and 2.75, but it is possible to use other values.

`ofilename`
: Output file name.

Details

The `rolwinmulcor_1win` function estimates the rolling window correlation among multiple time series (multi-variate case) sampled on identical time points for ONLY ONE window-length and plots the rolling correlation coefficients and their respective p-values. `rolwinmulcor_1win` uses the functions *rollapply* (package:zoo) that is able to tackle matrices, the native R function *p.adjust* (package:stats), and some pieces of code and an auxiliary function that we have created specifically for our function `rolwinmulcor_1win` and *RolWinMulCor* R package.

Value

Outputs:

Plot output: *screen* or *'plot_multivariate_'* + *'ofilename* + *.png, .jpg, .eps* or *.pdf'*.

Numerical output: a list containing *Correlation_coefficients*, *P_values_corrected*, and *P_values_not_corrected*, which are self-explanatory.

Author(s)

Josué M. Polanco-Martínez (a.k.a. jomopo).
DeustoTech - Deusto Institute of Technology,
Faculty of Engineering, University of Deusto,
Avda. Universidades, 24, Bilbao, SPAIN.
Web1: https://scholar.google.es/citations?user=8djLIhcAAAAJ&hl=en.
Web2: http://www.researchgate.net/profile/Josue_Polanco-Martinez.
Email: <josue.m.polanco@gmail.com>, <josue.polanco@deusto.es>

References

Abdi H. Multiple correlation coefficient, in Encyclopedia of Measurement and Statistics, N. J. Salkind, Ed. Sage, Thousand Oaks, CA, USA, 2007; 648-651.
<URL: https://personal.utdallas.edu/~herve/Abdi-MCC2007-pretty.pdf>.

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B, 57 (1), 289-300. <URL: https://rss.onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1995.tb02031.x>.

Polanco-Martínez, J. M. (2019). Dynamic relationship analysis between NAFTA stock markets using nonlinear, nonparametric, non-stationary methods. Nonlinear Dynamics, 97(1), 369-389. <URL: https://doi.org/10.1007/s11071-019-04974-y>.

Examples

```
# Loading packages
library("RolWinMulCor")
library("zoo")
library("pracma")
# Loading data set
data(synthetic_data)
# Testing the function 'rolwinmulcor_1win' (multi-variate case)
# Window length = 51 and plot output in screen format
test1_rolwinmulcor_1win <- rolwinmulcor_1win(synthetic_data, varnametsY="Y",
                           varnametsX=paste("X1", "X2", "X3", sep=", "),
                           coltsX = c("red", "blue", "green"), units="NU",
                           parcen=c(0.6,25), widthwin=51, device="screen")
#  Window length = 75 and plot output in screen format
```

```
test2_rolwinmulcor_1win <- rolwinmulcor_1win(synthetic_data, varnametsY="Y",
                           varnametsX=paste("X1", "X2", "X3", sep=", "),
                           coltsX = c("red", "blue", "green"), units="NU",
                           parcen=c(0.6,25), widthwin=75, device="screen")
```

`rolwinmulcor_heatmap` *Estimate the Rolling Window Correlation for the multi-variate case and plot the results as a heat map.*

Description

The `rolwinmulcor_heatmap` function estimates the rolling (running) window correlation among several time series (multi-variate case) sampled on identical time points for all the possible (i.e. from 3 to the number of elements of the time series under analysis) window-lengths or for a band of window-lengths and plots the correlation coefficients and their respective p-values (<= 0.05) as a heat map. The multivariate case is based on the concept of multiple regression and generalizes the standard coefficient of correlation. We follow and extend the work of Abdi (2007) to implement computationally this technique. The `rolwinmulcor_heatmap` function is highly flexible since this contains a great number of parameters to control the estimation of correlation and features of the plot output. A list of parameters are described in the following lines.

Usage

```
rolwinmulcor_heatmap(inputdata, varnametsY="Y", varnametsX="", units="",
                     coltsY="black", coltsX="", LWDtsY=1, LWDtsX="",
                     CEXLAB=1.15, CEXAXIS=1.5, NUMLABX=5, parcen=c(0.5,25),
                     rmltrd="Y", Scale="Y", typewidthwin="FULL",
                     widthwin_1=3, widthwin_N=dim(inputdata)[1],
                     Align="center", pvalcorectmethod="BH", device="screen",
                     Hfig=900, Wfig=900, resfig=150, Hpdf=7, Wpdf=7, ofilename)
```

Arguments

`inputdata`	Matrix of P columns: time, dependent variable (Y), and independent variables ($X_1, X_2, ..., X_{P-2}$).
`varnametsY`	Name of the dependent variable: Y. Please note that the name of this variable MUST be defined.
`varnametsX`	Name of the independent variables: $X_1, X_2, ..., X_{p-2}$. Please note that the names of these variables MUST be defined in this way: varnametsX=c("X1", "X2",..., sep=", ").
`units`	Time's unit (e.g. days, weeks, years, etc.) for the variables under analysis. Please note that the units MUST be defined.
`coltsY, coltsX`	The colors to be used when the dependent (Y) and independent ($X_1, X_2, ...$) variables are plotted, by default for the dependent variable the color is "black" (although other colors can be used) and the colors for the independent variables MUST be provided (e.g. coltsX=c("red","blue",...)).

`LWDtsY, LWDtsX` These parameters are used to define the line-widths when the dependent and independent variables are plotted, by default and for the dependent variable this has a values of 1 (although other values (widths) can be used), please note that for the independent variables the line-widths MUST be provided (e.g. LWDtsX = c(1,2,...)).

`CEXLAB, CEXAXIS`
These parameters are used to plot the sizes of the X-axis and Y-axis labels and X- and Y-axis, by default these parameters have values of 1.15 and 1.5, respectively, but other values can be used.

`NUMLABX` Number of labels for (all) the X's axis, by the default is 5, but it is possible to use other values.

`parcen` These parameters contain two values: the first one is to control the position of the title, by default it is 0.5, but you should try with other close values to obtain the title centered, e.g. 0.4 or 0.8 (please avoid to use large values); the second value is to define the spaces between the names of variables, by default is 25 spaces, but you could try other values to fit properly the names of variables in the title. We use "mtext" to produce the title (please loot at R>?mtext for more information).

`rmltrd` Remove (by default is "Y" or "y"; please use "N" or "n" otherwise) the linear trend in the time series under analysis.

`Scale` Scale (by default is "Y" or "y"; please use "N" or "n" otherwise) is used to "normalize" or "standardize"the time series under analysis.

`typewidthwin` There are two options to estimate the rolling window correlations and to plot the heat map: (1) typewidthwin="PARTIAL" to plot a band of windows from widthwin_1 to widthwin_N (both parameters were defined previously), or (2) typewidthwin="FULL" to plot all the possible window-lengths (from 3 to dim(inputdata)[1]), please note that typewidthwin by default is FULL.

`widthwin_1` First value for the size (length) of the windows when the option typewidthwin="PARTIAL" is selected, the minimum value is 3 (the default value), but you should define this parameter (please note that widthwin_1 < widthwin_N).

`widthwin_N` Last value for the size (length) of the windows when the option typewidthwin="PARTIAL" is selected, by default is dim(inputdata)[1], but you should define this parameter (please note that widthwin_1 < widthwin_N).

`Align` To align the rolling object, RolWinMulCor ONLY uses the "center" option (please look at: R>?rollapply) to ensure that variations in the correlation are aligned with the variations in the relationship of the time series under study rather than being shifted (Polanco-Martínez 2019).

`pvalcorectmethod`
The p-value correction method to be used, by default the method of Benjamini and Hochberg (BH) (1995) is used since this is less conservative and performs much better than Bonferroni, but other six methods (e.g. Holm, Bonferroni, etc.) are available (please look at: R>?p.adjust).

`device` Kind of plot output (please look at: R>?device), there are five options: "png", "jpeg/jpg", "eps", "pdf" and "screen." By default device is "screen."

`Hfig, Wfig`	Plot's height and width (for the device) for "png" and "jpg" format (look at R>?png or R>?jpg), by default Hfig and Wfig have values of 900, but other values can be used.
`resfig`	Image resolution (in "ppi") for the plot in "png" and "jpg" format (look at R>?png , R>?jpg or R>?jpeg), by default this has a value of 150, but other image resolutions can be used.
`Hpdf, Wpdf`	Plot's height and width (for the device) for "pdf" or "eps" format (>R?pdf or >R?postscript), by default Hpdf and Wpdf are equal to 7, but it is possible to use other values.
`ofilename`	Output file name.

Details

The `rolwinmulcor_heatmap` function estimates the rolling window correlation between multiple time series (multi-variate case) sampled on identical time points for all the possible window-lengths or for a band of window-lengths and plots the rolling correlation coefficients and their respective p-values as a heat map. `rolwinmulcor_heatmap` uses the functions *rollapply* (package:zoo) that is able to tackle matrices, the native R function *p.adjust* (package:stats), and some pieces of code and an auxiliary function that we have created specifically for our function `rolwinmulcor_heatmap` and *RolWinMulCor* R package.

Value

Outputs:

Plot output (heat map): *screen* or *'heatmap_multivariate_'* + *'ofilename* + *.png, .jpg, .eps* or *.pdf'*.

Numerical output: two lists *matcor* and *pvalscor* containing the correlation matrix and their corresponding corrected p-values, *NoWindows* and *Windows* that contains the number of windows and the window-lengths.

Author(s)

Josué M. Polanco-Martínez (a.k.a. jomopo).
DeustoTech - Deusto Institute of Technology,
Faculty of Engineering, University of Deusto,
Avda. Universidades, 24, Bilbao, SPAIN.
Web1: https://scholar.google.es/citations?user=8djLIhcAAAAJ&hl=en.
Web2: http://www.researchgate.net/profile/Josue_Polanco-Martinez.
Email: <josue.m.polanco@gmail.com>, <josue.polanco@deusto.es>

References

Abdi H. Multiple correlation coefficient, in Encyclopedia of Measurement and Statistics, N. J. Salkind, Ed. Sage, Thousand Oaks, CA, USA, 2007; 648-651.
<URL: https://personal.utdallas.edu/~herve/Abdi-MCC2007-pretty.pdf>.

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B, 57 (1),

289-300. <URL: https://rss.onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1995.tb02031.x>.

Polanco-Martínez, J. M. (2019). Dynamic relationship analysis between NAFTA stock markets using nonlinear, nonparametric, non-stationary methods. Nonlinear Dynamics, 97(1), 369-389. <URL: https://doi.org/10.1007/s11071-019-04974-y>.

Examples

```
# Loading packages
library("RolWinMulCor")
library("zoo")
library("pracma")
# Loading data set
data(synthetic_data)
# Testing the function rolwinmulcor_heatmap
# typewidthwin="PARTIAL," window lengths from 21 to 31 and plot output in screen format
test1_rolwinmulcor_heatmap <- rolwinmulcor_heatmap(synthetic_data, varnametsY="Y",
                              varnametsX=c("X1", "X2", "X3"),
                              coltsX=c("red", "blue", "green"), units="NU",
                              parcen=c(0.75,25), typewidthwin="PARTIAL",
                              widthwin_1=21, widthwin_N=31, device="screen")
# This example could takes a long time!
# typewidthwin="FULL" and pot output in PDF format

test2_rolwinmulcor_heatmap <- rolwinmulcor_heatmap(synthetic_data, varnametsY="Y",
                              varnametsX=c("X1", "X2", "X3"),
                              coltsX = c("red", "blue", "green"), units="NU",
                              parcen=c(0.75,25), typewidthwin="FULL", device="pdf",
                              ofilename="test2")
```

`synthetic_data` *Synthetic data set to test the functions of* RolWinMulCor.

Description

The data set `synthetic_data` contains five columns, the first one is the time and the next four columns are sinusoidal time series that contains two periodical signals (at 11 and 51, with a phase of zero and amplitude of 1 –please note that these quantities are dimensionless) 'contaminated' by Gaussian noise (with mean of 0 and standard deviation of 0.25) for the intervals 1-100 (signal at 11) and 200-400 (signal at 51) and Gaussian noise (with mean of 0 and standard deviation of 1) otherwise.

Usage

```
data(synthetic_data)
```

Format

One file in ASCII format and columns are separated by spaces.

Source

Author's own production (Josué M. Polanco-Martínez).

`YX_ecological_data` *Ecological data set to test the functions of* RolWinMulCor.

Description

The data set `YX_ecological_data` contains four columns, the first one ("Years") is the time (years from 1700 to 1936), the second is the first component principal ("PC1") of the reconstructed Atlantic Bluefin Tuna (BFT) captures (Ganzedo et al. 2016, Polanco-Martínez et al. 2018), the third are reconstructions of sea surface temperature ("SST") from the Northern Hemisphere (NH) (Mann et al. 2009), and the fourth column contains reconstructions of total solar irradiance ("TSI") (Lean 2000).

Usage

```
data(YX_ecological_data)
```

Format

One file in ASCII format and columns are separated by spaces.

Source

Ganzedo, U., Polanco-Martínez, J. M., Caballero-Alfonso, Á. M., Faria, S. H., Li, J., Castro-Hernández, J. J. (2016). Climate effects on historic bluefin tuna captures in the Gibraltar Strait and Western Mediterranean. Journal of Marine Systems, 158, 84-92. <URL: https://doi.org/10.1016/j.jmarsys.2016.02.002>.

Lean, J. (2000). Evolution of the Sun's spectral irradiance since the Maunder Minimum. Geophysical Research Letters, 27(16), 2425-2428. <URL: https://doi.org/10.1029/2000GL000043>. Lean Web TSI data set: <URL: https://www.ncdc.noaa.gov/paleo-search/study/5788>.

Mann, M.E., Zhang, Z., Rutherford, S., Bradley, R.S., Hughes, M.K., Shindell, Ammann, G., Faluvegi, G., Ni, F. (2009). Global signatures and dynamical origins of the little ice age and medieval climate anomaly. Science 326, 1256-1260. <URL: https://doi.org/10.1126/science.1177303>. Mann et al. Web SST data set: <URL: http://www.meteo.psu.edu/holocene/public_html/supplements/MultiproxySpatial09/results/nhscr>.

Polanco-Martínez, J. M., Caballero-Alfonso, A. M., Ganzedo, U., Castro-Hernández, J. J. (2018). A reconstructed database of historic bluefin tuna captures in the Gibraltar Strait and Western Mediterranean. Data in Brief, 16, 206-210. <URL: https://doi.org/10.1016/j.dib.2017.11.028>.

Index