Package ‘TSS.RESTREND’

October 12, 2022

Type Package

Title Time Series Segmentation of Residual Trends

Version 0.3.1

Description
Time Series Segmented Residual Trends is a method for the automated detection of land degradation from remotely sensed vegetation and climate datasets. TSS-RESTREND incorporates aspects of two existing degradation detection methods: RESTREND which is used to control for climate variability, and BFAST which is used to look for structural changes in the ecosystem. The full details of the testing and justification of the TSS-RESTREND method (version 0.1.02) are published in Burrell et al., (2017). <doi:10.1016/j.rse.2017.05.018>. The changes to the method introduced in version 0.2.03 focus on the inclusion of temperature as an additional climate variable. This allows for land degradation assessment in temperature limited drylands. A paper that details this work is currently under review. There are also a number of bug fixes and speed improvements. Version 0.3.0 introduces additional attribution for eCO2, climate change and climate variability the details of which are in press in Burrell et al., (2020). The version under active development and additional example scripts showing how the package can be applied can be found at <https://github.com/ArdenB/TSSRESTREND>.

License GPL-3

LazyData TRUE

Imports stats, graphics, utils, bfast (>= 1.5.7), broom, strucchange, ggplot2, RcppRoll, mblm

Suggests rgl, car

Depends R (>= 3.6.0)

RoxygenNote 7.1.1

NeedsCompilation no

Author Arden Burrell [aut, cre]

Maintainer Arden Burrell <arden.burrell@gmail.com>

Repository CRAN

Date/Publication 2020-08-02 20:10:02 UTC

R topics documented:

`TSS.RESTREND-package` *TSS.RESTREND: Time Series Segmented RESidual TREND*

Description

The TSS-RESTREND package is designed used via the TSS-RESTREND function which calls all the other functions in the correct order to perform a complete TSS-RESTREND analysis. The individual functions have been made callable to allow for greater user control. The version in active development, as well as additional documentation, data and example scripts, see https://github.com/ArdenB/TSSRESTREND.

Details

Version 0.3.0

Note

This code is a demostration of the method and not the code used in Burrell et. al., (2017). The original code uses both python and R and was designed for batch proccessing. For the pixels tested using the TSS.RESTREND package it produces identical results.

Author(s)

Maintainer: Arden Burrell <arden.burrell@gmail.com>

`ACP.calculator` *Antecedental Rainfall (and temperature) Accumulation calculator for the VI Complete Time Series*

Description

Takes the Complete Time Series Vegetation Index and a table of every possible accumulation period and offset period for precipitation and temperature(optional). A OLS is calculated `lm` for every combination of VI ~ rainfall (and temperature if that is included). if only the VPR is being calculated, this Function preferences those results where slope>0 (increase in rainfall causes an increase in vegetation),returning the rainfall accumulation that has the highest R-squared and a positive slope. If no combination produces a positive slope then the one with the highest R-squared is returned.

Usage

```
ACP.calculator(
  CTSR.VI,
  ACP.table,
  ACT.table = NULL,
  allow.negative = FALSE,
  allowneg.retest = FALSE
)
```

Arguments

`CTSR.VI`	Complete Monthly Time Series of Vegetation Index values. An object of class `'ts'` object without NA's.
`ACP.table`	A table of every combination of offset period and accumulation period.for precipitation ACP.table can be calculated using the `climate.accumulator`.
`ACT.table`	A table of every combination of offset period and accumulation period.for temperature ACP.table can be calculated using the `climate.accumulator`.
`allow.negative`	If true, will not preference positive slope in either CTSR or VI calculations. default=FALSE is set because negative associations between rainfall and vegetation in water limited ecosystems is unexpected If temperature data is included then this paramter is forced to TRUE.
`allowneg.retest`	default=FALSE If temperature data is provided but found to not be significant then a retest is performed. This paramter is to allow negative on re-test.

Value

A list containing:

summary a Matrix containing "slope", "intercept", "p.value", "R^2.Value", "Break.Height", "Slope.Change" of the `lm` between Antecedental Rainfall Accumulation (CTSR.RF) and the CTSR.VI

CTSR.precip see CTSR.RF in `TSSRESTREND` for description

CTSR.osp The offest period for the complete time series rainfall

CTSR.acp The accumulation period for the complete time series rainfall

CTSR.tmp The optimally accumulated CTS temperature

CTSR.tosp The offest period for the complete time series temperature

CTSR.tacp The accumulation period for the complete time series temperature

Examples

```
#Find the data
vi.path <- system.file("extdata", "rabbitVI.csv", package = "TSS.RESTREND", mustWork = TRUE)
in.VI <- read.csv(vi.path)
CTSR.VI <- ts(in.VI, start=c(1982, 1), end=c(2013,12), frequency = 12)
data(rabbitACPtable)
ACPres <- ACP.calculator(CTSR.VI, rabbitACPtable)
print(ACPres$summary)
```

AnMaxVI *Annual max VI Calculator*

Description

Takes the montly time series of the VI and calculates the growing season max VI. In series where the peak occurs in November or December, an interannual growing season is assessed.

Usage

```
AnMaxVI(CTSR.VI)
```

Arguments

`CTSR.VI`	Complete Monthly Time Series of Vegetation Index values. An object of class `'ts'` object without NA's.

Value

Max(anu.VI) The annual (Growing season) max VI. See `TSSRESTREND`

Max.Month The month number where max values were observed (1 for January). if month > 12, the peak was detected in Nov, Dec, Jan. In this case the peak seasonal value and position is used.

index(VI.index) the index of the CTSR.VI ts that the anu.VI values occur at. See `TSSRESTREND`. Note.R indexs from 1 rather than 0.

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
anmax <- AnMaxVI(stdRESTRENDCTSR$cts.NDVI)
print(anmax)
```

AnnualClim.Cal — *Antecedental accumulation calculator for the annual max VI time series*

Description

Takes the Annual Max VI Time Series, the VI.index and tables of every possible accumulation period and offset period for preciptation and Temperature (optional). A OLS is calculated `lm` for every combination of VI ~ rainfall. If temperature is provided The formula is (VI ~ rainfall + temperature). By defualt, this function preferences those results where slope>0 (increase in rainfall causes an increase in vegetation), returning the rainfall accumulation that has the highest R-squared and a positive slope. If no combinations produce a positive slope then the one with the highest Rsquared is returned. TO DO: non peramtric and other variables

Usage

```
AnnualClim.Cal(
  anu.VI,
  VI.index,
  ACP.table,
  ACT.table = NULL,
  Breakpoint = FALSE,
  allow.negative = FALSE,
  allowneg.retest = FALSE
)
```

Arguments

`anu.VI`	The annual (Growing season) max VI. Must be a object of class `'ts'` without NA's. if anu.VI=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`VI.index`	the index of the CTSR.VI ts that the anu.VI values occur at. Must be the same length as anu.VI. NOTE. R indexs from 1 rather than 0. if VI.index=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`ACP.table`	A table of every combination of offset period and accumulation period.for precipitation ACP.table can be calculated using the `climate.accumulator`.
`ACT.table`	A table of every combination of offset period and accumulation period.for temperature ACP.table can be calculated using the `climate.accumulator`.

`Breakpoint` Used when calcualting rf.bf and rf.af for ts with breakpoints in the VPR. See `CHOW`

`allow.negative` If true, will not preference positive slope in either CTSR or VI calculations. default=FALSE is set because negative associations between rainfall and vegetation in water limited ecosystems is unexpected If temperature data is included then this paramter is forced to TRUE.

`allowneg.retest` default=FALSE If temperature data is provided but found to not be significant then a retest is performed. This paramter is to allow negative on re-test.

Value

summary a Matrix containing "slope", "intercept", "p.value", "R^2.Value", "Break.Height", "Slope.Change" of the `lm` of VI ~ rainfall. If Breakpoint, summary covers both rf.b4 and rf.af.

acu.RF (aka. annual.precip)The optimal accumulated rainfall for anu.VI. Mut be a object of class 'ts' and of equal length to anu.VI. It is caculated from the ACP.table by finding the acp and osp that has the largest R^2 value. `lm`(anu.VI ~ rainfall)

acu.TM (aka, annual.temp) The optimal accumulated rainfall for anu.T<. Mut be a object of class 'ts' and of equal length to anu.VI. It is caculated from the ACT.table by finding the tacp and tosp that has the largest R^2 value. `lm`(anu.VI ~ rainfall+temperature)

rf.b4 The optimal acumulated rainfall before the Breakpoint

rf.af The Optimally accumulated rainfall after the Breakpoint

tm.b4 The optimal acumulated temperature before the Breakpoint

tm.af The Optimally accumulated temperature after the Breakpoint

osp The offest period for the annual max time series rainfall

acp The accumulation period for the annual max time series rainfall

tosp The offest period for the annual max time series temperature

tacp The accumulation period for the annual max time series temperature

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
ARC <- AnnualClim.Cal(stdRESTREND$max.NDVI, stdRESTREND$index, stdRESTRENDrfTab)
print(ARC)
## Not run:

#Test the complete time series for breakpoints
VPRBFdem <- VPR.BFAST(segVPRCTSR$cts.NDVI, segVPRCTSR$cts.precip)
bp<-as.numeric(VPRBFdem$bkps)

#test the significance of the breakpoints
reschow <- CHOW(segVPR$max.NDVI, segVPR$acum.RF, segVPR$index, bp)
brkp <- as.integer(reschow$bp.summary["yr.index"])
```

```
ARCseg <-AnnualClim.Cal(segVPR$max.NDVI, segVPR$index, segVPRrfTab, Breakpoint = brkp)
print(ARCseg)

## End(Not run)
```

CHOW *Chow test on detected breakpoints*

Description

Takes the breakpoints detected by VPR.BFAST, finds the most significant one then tests it in both the residuals and the VPR.

Usage

```
CHOW(anu.VI, acu.RF, VI.index, breakpoints, acu.TM = NULL, sig = 0.05)
```

Arguments

anu.VI	The annual (Growing season) max VI. Must be a object of class 'ts' without NA's. if anu.VI=FALSE, it will be calculated from the CTSR.VI using AnMaxVI.
acu.RF	The optimal accumulated rainfall for anu.VI. Must be a object of class 'ts' without NA's and be of equal length and temporal range to anu.VI. if anu.RF=FALSE, it will be calculated from ACP.table usingthe AnnualClim.Cal
VI.index	the index of the CTSR.VI ts that the anu.VI values occur at. Must be the same length as anu.VI. NOTE. R indexs from 1 rather than 0. if VI.index=FALSE, it will be calculated from the CTSR.VI using AnMaxVI.
breakpoints	vector containing the breakpoints detected by VPR.BFAST (bkps)
acu.TM	The optimal accumulated rainfall for anu.TM. Must be a object of class 'ts' without NA's and be of equal length and temporal range to anu.TM. if anu.TM=FALSE, it will be calculated from ACT.table usingthe AnnualClim.Cal
sig	Significance of all the functions. defualt sig=0.05

Value

n.Method The method that the ts should be tested with. TSSRESTREND internal communication.

bp.summary Summary of the most signifcant breakpoint in the residuals and VPR. see sctest

allbp.index the Annual index of every breakpoint. Used by plot.TSSRESTREND

bpRESID.chow Chow test in the VPR residuals. See sctest

bpVPR.chow Chow test in the VPR. See sctest

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
## Not run:
#Test the complete time series for breakpoints
VPRBFdem <- VPR.BFAST(segVPRCTSR$cts.NDVI, segVPRCTSR$cts.precip)
bp <- as.numeric(VPRBFdem$bkps)
#test the significance of the breakpoints
reschow <- CHOW(segVPR$max.NDVI, segVPR$acum.RF, segVPR$index, bp)
print(reschow)

## End(Not run)
```

`climate.accumulator` *Climate Accumulator*

Description

Takes the time series of rainfall and returns a rainfall accumulation table of every possible combination of the max accumulation period and the max offset period.

Usage

```
climate.accumulator(
  CTSR.VI,
  clim.data,
  max.acp,
  max.osp,
  temperature = FALSE,
  cliwindow = 0
)
```

Arguments

`CTSR.VI`	Complete Time Series of Vegetation Index. An object of class `'ts'`. Monthly time series of VI values
`clim.data`	Complete Time Series of monthly rainfall or temperature. An object of class `'ts'`. Must have the same end date as CTSR.VI and be longer than the CTSR.VI by more than the max acsumuation period (max.acp) plus the max offset period.(max.ops)
`max.acp`	The max accumuation period. Must be an integer > 1.
`max.osp`	The max offset period. Must be an integer >1
`temperature`	Bool. If the clim.data being accumulated is temperature, will take a mean not a sum. This makes it easier to comapare regions with different accumulation and offset periods. This is new in v0.3.0. Defualt=FALSE which replicates the behaviour of TSSRESTREND versions <0.2.16.
`cliwindow`	The size of the window in years to be used for calculating climate change.

Value

ACP.table A matrix with ever possible accumuated climate combination

Examples

```
# Define the max accumuulation period
acp <- 12
#Define the max offset period
osp <- 4
rftable <- climate.accumulator(segRESTRENDCTSR$cts.NDVI, segRESTRENDctRF$precip, acp, osp)
```

`franksCO2` *Franks CO2 Vegetation correction*

Description

Adjusts a vegetation time series to account for CO2 fertilisation

Usage

```
franksCO2(CTSR.VI, C4frac, CO2 = FALSE, refyear = 1980)
```

Arguments

`CTSR.VI`	Complete Monthly Time Series of Vegetation Index values. An object of class `'ts'` object without NA's.
`C4frac`	The fraction of vegetation that follows the C4 photosynthetic pathway, between 0 and 1
`CO2`	A timeseries containg the CO2 concentration. The defualt is CMIP5 RCP8.5
`refyear`	The Year that acts as a baseline for CO2. All vegetation values will be scaled # to the CO2 concentration of this year. Defulat is 1980. THis function will pick the first value in the selected year,

Value

CTSR.VIadj A version of the CTSR.VI data that has been adjusted to account for CO2

Author(s)

Arden Burrell, aburrell@whrc.org

plot.TSSRESTREND *Plot Function for ojects of the TSSRESTREND class*

Description

Produces plots for class TSSRESTREND

Usage

```
## S3 method for class 'TSSRESTREND'
plot(x, plots = "all", sig = 0.05, ...)
```

Arguments

x	Object of class TSSRESTREND
plots	Defualts to "all", will produce the standard plots, plots can be called individually, "bfast", "chow", "VPR", "anu.VI", "final".
sig	Significance
...	further arguments passed to the function.

print.TSSRESTREND *Print function for class TSSRESTREND*

Description

print function for objects of class TSSRESTREND

Usage

```
## S3 method for class 'TSSRESTREND'
print(x, ...)
```

Arguments

x	Object of class TSSRESTREND
...	further arguments passed to the function.

rabbitACPtable — *Rabbit Impacted Vegetation Precipitation Accumulation Table*

Description

Rainfall Accumulation Table form a part of the Simpson-sStrzelecki Dunefields bioregion impacted by reduced rabbit predation after the release of RHD.

Usage

```
rabbitACPtable
```

Format

A matrix containing the complete time series of every acp and ofp. See `climate.accumulator` for details

Source

Awap data from `http://www.csiro.au/awap/cgi/awap2.pl`

Examples

```
data(rabbitACPtable)
```

RESTREND — *RESTREND (RESidual TREND)*

Description

For ts with no significant breakpoints in the residuals or the VPR. Takes annual VI max and it associated optimal accumulated precipitation

Usage

```
RESTREND(
  anu.VI,
  acu.RF,
  VI.index,
  acu.TM = NULL,
  sig = 0.05,
  retnonsig = FALSE
)
```

Arguments

`anu.VI`	The annual (Growing season) max VI. Must be a object of class 'ts' without NA's. if anu.VI=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`acu.RF`	The optimal accumulated rainfall for anu.VI. Must be a object of class 'ts' without NA's and be of equal length and temporal range to anu.VI. if anu.RF=FALSE, it will be calculated from ACP.table usingthe `AnnualClim.Cal`
`VI.index`	the index of the CTSR.VI ts that the anu.VI values occur at. Must be the same length as anu.VI. NOTE. R indexs from 1 rather than 0. if VI.index=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`acu.TM`	The optimal accumulated rainfall for anu.TM. Must be a object of class 'ts' without NA's and be of equal length and temporal range to anu.TM. if anu.TM=FALSE, it will be calculated from ACT.table usingthe `AnnualClim.Cal`
`sig`	Significance of all the functions. defualt sig=0.05
`retnonsig`	Bool. New in v0.3.0. Allows TSSRESTREND to return change estimates of values that filed the sig component in the residual analysis. defualt FALSE will give the same result as eralier versions.

Value

a list of class TSSRESTREND. See `TSSRESTREND` for details. Note. if called seperatly from TSS-RESTREND, this list will be incomplete.

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
restrend <- RESTREND(stdRESTREND$max.NDVI, stdRESTREND$acc.precip, stdRESTREND$index)
print(restrend)
```

`seg.RESTREND` *Segmented RESTREND (RESidual TREND)*

Description

For ts with a significant breakpoints in the residuals but not in the VPR. Takes annual VI max and it associated optimal accumulated precipitation and regresses it with a dummy variable that is 0 before the breakpoint and 1 after it

Usage

```
seg.RESTREND(
  anu.VI,
  acu.RF,
  VI.index,
  breakpoint,
  acu.TM = NULL,
  sig = 0.05,
  retnonsig = FALSE
)
```

Arguments

`anu.VI`	The annual (Growing season) max VI. Must be a object of class 'ts' without NA's. if anu.VI=FALSE, it will be calculated from the CTSR.VI using AnMaxVI.
`acu.RF`	The optimal accumulated rainfall for anu.VI. Must be a object of class 'ts' without NA's and be of equal length and temporal range to anu.VI. if anu.RF=FALSE, it will be calculated from ACP.table usingthe AnnualClim.Cal
`VI.index`	the index of the CTSR.VI ts that the anu.VI values occur at. Must be the same length as anu.VI. NOTE. R indexs from 1 rather than 0. if VI.index=FALSE, it will be calculated from the CTSR.VI using AnMaxVI.
`breakpoint`	The index of the most significant breakpoint as determined using CHOW.
`acu.TM`	The optimal accumulated rainfall for anu.TM. Must be a object of class 'ts' without NA's and be of equal length and temporal range to anu.TM. if anu.TM=FALSE, it will be calculated from ACT.table usingthe AnnualClim.Cal
`sig`	Significance of all the functions. defualt sig=0.05
`retnonsig`	Bool. New in v0.3.0. Allows TSSRESTREND to return change estimates of values that filed the sig component in the residual analysis. defualt FALSE will give the same result as eralier versions.

Value

a list of class TSSRESTREND. See TSSRESTREND for details. Note. if called seperatly from TSS-RESTREND, this list will be incomplete.

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
# brkp can be determined using VPR.BFAST and CHOW.
brkp <-  as.integer(11)
resu <- seg.RESTREND(segRESTREND$max.NDVI, segRESTREND$acc.precip, segRESTREND$index, brkp)
```

seg.VPR *Segmented Vegetation Climate Relationship*

Description

For a ts with a significant breakpoints in the the VPR/VCR. This function takes annual VI max, the optimal accumulated precipitation (& temperature) before and after the breakpoint, then caculates the Standard Variance of the climate cariables.Theen an OLS is performed with a dummy variable to reperesent the breakpoint (0 before the breakpoint and 1 after it)..

Usage

```
seg.VPR(
  anu.VI,
  acu.RF,
  VI.index,
  breakpoint,
  rf.b4,
  rf.af,
  acu.TM = NULL,
  tm.b4 = NULL,
  tm.af = NULL,
  sig = 0.05,
  retnonsig = FALSE
)
```

Arguments

`anu.VI`	The annual (Growing season) max VI. Must be a object of class `'ts'` without NA's. if anu.VI=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`acu.RF`	The optimal accumulated rainfall for anu.VI. Must be a object of class `'ts'` without NA's and be of equal length and temporal range to anu.VI. if anu.RF=FALSE, it will be calculated from ACP.table usingthe `AnnualClim.Cal`
`VI.index`	the index of the CTSR.VI ts that the anu.VI values occur at. Must be the same length as anu.VI. NOTE. R indexs from 1 rather than 0. if VI.index=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`breakpoint`	The index of the most significant breakpoint as determined using `CHOW`.
`rf.b4`	If a breakpoint in the VPR is detected this is the optimial accumulated rainfall before the breakpoint. must be the same length as the anu.VI. If ACP.table is provided it will be generated using `AnnualClim.Cal`
`rf.af`	If a breakpoint in the VPR is detected this is the optimial accumulated rainfall after the breakpoint. must be the same length as the anu.VI. If ACP.table is provided it will be generated using `AnnualClim.Cal`
`acu.TM`	The optimal accumulated rainfall for anu.TM. Must be a object of class `'ts'` without NA's and be of equal length and temporal range to anu.TM. if anu.TM=FALSE, it will be calculated from ACT.table usingthe `AnnualClim.Cal`

tm.b4
: If a breakpoint in the VCR is detected this is the optimial accumulated temperature before the breakpoint. It must be the same length as the anu.VI. If ACT.table is provided it will be generated using AnnualClim.Cal

tm.af
: If a breakpoint in the VCR is detected this is the optimial accumulated temperature after the breakpoint. It must be the same length as the anu.VI. If ACT.table is provided it will be generated using AnnualClim.Cal

sig
: Significance of all the functions. defualt sig=0.05

retnonsig
: Bool. New in v0.3.0. Allows TSSRESTREND to return change estimates of values that filed the sig component in the residual analysis. defualt FALSE will give the same result as eralier versions.

Value

a list of class TSSRESTREND. See TSSRESTREND for details. Note. if called seperatly from TSS-RESTREND, this list will be incomplete.

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
brkp <- as.integer(24) #calculated using th CHOW (DONTRUN) example
VPRres <- seg.VPR(segVPR$max.NDVI, segVPR$acum.RF, segVPR$index, brkp, segVPR$RFB4, segVPR$RFAF)
print(VPRres)
```

---

segRESTREND *Data frame containing the annual data for a segRESTREND analysis*

Description

contains anu.VI, acu.RF, VI.index, rf.b4, rf.af. Range 1982 - 2013. Breakpoint for this pixel is 24 (2005)

Usage

```
segRESTREND
```

Format

R data frame

Source

gimms-package

`segRESTRENDctRF` *Data frame containing the raw rainfall data set for the segRESTREND data, ending dec 2013 with a frquency of 12*

Description

contains raw monthly precipitation

Usage

```
segRESTRENDctRF
```

Format

R data frame

Source

Awap data from http://www.csiro.au/awap/cgi/awap2.pl

See Also

`stdRESTREND` for the annual values for the same pixel

`segRESTRENDCTSR` *Data frame containing the Complete Times Series data for a segmented RESTREND analysis*

Description

contains CTSR.VI and CTSR.RF. See `TSSRESTREND`. Starts 1/1982 and extend to 12/2013

Usage

```
segRESTRENDCTSR
```

Format

R data frame

Source

a single pixel from `monthlyComposite`

See Also

`stdRESTREND` for the annual values for the same pixel

`segRESTRENDrfTab` *Precipitation Accumulation Table for the segRESTREND demonstration pixel*

Description

Rainfall Accumulation Table

Usage

```
segRESTRENDrfTab
```

Format

A matrix containing the complete time series of every acp and ofp. See `climate.accumulator` for details

Source

Awap data from `http://www.csiro.au/awap/cgi/awap2.pl`

`segVPR` *Data frame containing the annual data for a segVPR analysis*

Description

contains anu.VI, acu.RF, VI.index, rf.b4, rf.af. Range 1982 - 2013. Breakpoint for this pixel is 24 (2005)

Usage

```
segVPR
```

Format

R data frame

Source

`gimms-package`

---

segVPRctRF *Data frame containing the raw rainfall data set for the segVPR data, ending dec 2013 with a frquency of 12*

Description

contains raw monthly precipitation

Usage

```
segVPRctRF
```

Format

R data frame

Source

Awap data from http://www.csiro.au/awap/cgi/awap2.pl

See Also

`stdRESTREND` for the annual values for the same pixel

---

segVPRCTSR *Data frame containing the Complete Times Series data for a segmented VPR (VEGETATION PRECIPTATION RELATIONSHIP) analysis*

Description

contains CTSR.VI and CTSR.RF. See `TSSRESTREND`. Starts 1/1982 and extend to 12/2013

Usage

```
segVPRCTSR
```

Format

R data frame

Source

a single pixel from `monthlyComposite`

See Also

`stdRESTREND` for the annual values for the same pixel

segVPRrfTab *Precipitation Accumulation Table for the segVPR demonstration pixel*

Description

Rainfall Accumulation Table

Usage

```
segVPRrfTab
```

Format

A matrix containing the complete time series of every acp and ofp. See `climate.accumulator` for details

Source

Awap data from `http://www.csiro.au/awap/cgi/awap2.pl`

stdRESTREND *Data frame containing the annual data for a standard Restrend analysis*

Description

contains anu.VI, acu.RF and VI.index. Range 1982 - 2013

Usage

```
stdRESTREND
```

Format

R data frame

Source

`gimms-package`

`stdRESTRENDctRF` *Data frame containing the raw rainfall data set ending dec 2013 with a frquency of 12*

Description

contains raw monthly precipitation

Usage

```
stdRESTRENDctRF
```

Format

R data frame

See Also

`stdRESTREND` for the annual values for the same pixel

`stdRESTRENDCTSR` *Data frame containing the Complete Times Series data for a standard Restrend analysis*

Description

contains CTSR.VI and CTSR.RF. See `TSSRESTREND`. Starts 1/1982 and extend to 12/2013

Usage

```
stdRESTRENDCTSR
```

Format

R data frame

Source

a single pixel from `monthlyComposite`

See Also

`stdRESTREND` for the annual values for the same pixel

`stdRESTRENDrfTab` *Precipitation Accumulation Table for the standard RESTREND demonstration pixel*

Description

Rainfall Accumulation Table

Usage

```
stdRESTRENDrfTab
```

Format

A matrix containing the complete time series of every acp and ofp. See `climate.accumulator` for details

Source

Awap data from http://www.csiro.au/awap/cgi/awap2.pl

`TSSRattribution` *Vegetation change attribution using the Time Series Segmentation of Residual Trends (MAIN FUNCTION)*

Description

This is a wrapper function around the TSS-RESTREND main function that dows additional attribution. It measues the Observed vegetation change, land use, climate change and climate varibility. Unlike TSSRESTREND function, this requires both temperature and precitation data fo work.

Usage

```
TSSRattribution(
  CTSR.VI,
  CTSR.RF,
  CTSR.TM,
  max.acp,
  max.osp,
  C4frac = 0,
  sig = 0.05,
  season = "none",
  exclude = 0,
  allow.negative = FALSE,
  allowneg.retest = FALSE,
  h = 0.15,
```

```
  returnmodels = FALSE,
  AnnualRes = FALSE,
  SkipError = TRUE,
  retnonsig = TRUE,
  splitclim = TRUE,
  cliwindow = 20,
  CO2 = FALSE,
  refyear = 1980
)
```

Arguments

`CTSR.VI`	Complete Monthly Time Series of Vegetation Index values. An object of class `'ts'` object without NA's.
`CTSR.RF`	Complete Time Series of Rainfall. An object of class 'ts' object without NA's and be the same length and cover the same time range as CTSR.VI. If ACP.table is provided, CTSR.RF will be automitaclly calculated using the `ACP.calculator`
`CTSR.TM`	Complete Time Series of temperature. An object of class 'ts' object without NA's and be the same length and cover the same time range as CTSR.VI. Default (CTSR.TM=NULL). If ACT.table is provided, CTSR.RF will be automitaclly calculated using the `ACP.calculator`
`max.acp`	The max accumuation period. Must be an integer > 1.
`max.osp`	The max offset period. Must be an integer >1
`C4frac`	The fraction of vegetation that follows the C4 photosynthetic pathway, between 0 and 1
`sig`	Significance of all the functions. defualt sig=0.05
`season`	See `bfast`. This season value only applies to bfast done using the CTS VPR. if a non VPR adjusted BFAST is performed.a harmonic season is used.
`exclude`	A numberic vector containg months excluded from breakpoint detection. This was included to allow sensor transitions to be masked.
`allow.negative`	If true, will not preference positive slope in either CTSR or VI calculations. default=FALSE is set because negative associations between rainfall and vegetation in water limited ecosystems is unexpected If temperature data is included then this paramter is forced to TRUE.
`allowneg.retest`	default=FALSE If temperature data is provided but found to not be significant then a retest is performed. This paramter is to allow negative on re-test.
`h`	See `bfast`, The.minimal segment size between potentially detected breaks in the trend model given as fraction relative to the sample size (i.e. the minimal number of observations in each segment divided by the total length of the timeseries. Default h = 0.15.
`returnmodels`	Return all the created models as well as the original data
`AnnualRes`	Report results in change per year. Defualt is False. Instead reports total change from the start to the end of the time series.

`SkipError`	Bool, If TRUE will handle most errors and return a dataframe filled with NA's. Usefull when processing large datasets to stop analysis failing on a single pixel. Use with caution. Defualt=TRUE
`retnonsig`	Bool. New in v0.3.0. Allows TSSRESTREND to return change estimates of values that filed the sig component in the residual analysis. defualt FALSE will give the same result as eralier versions.
`splitclim`	Bool, If TRUE Climate will be split into climate change and climate varibility as per Burrell et al., (2020). If FALSE. will just return climate as per IPCC: Chapter 3: Desertification in the IPCC Special Report on Climate Change, Desertification, Land Degradation, Sustainable Land Management, Food Security, and Greenhouse gas fluxes in Terrestrial Ecosystems. Defualt=True.
`cliwindow`	The size of the window in years to be used for calculating climate change.
`CO2`	A timeseries containg the CO2 concentration. The defualt is CMIP5 RCP8.5
`refyear`	The Year that acts as a baseline for CO2. All vegetation values will be scaled # to the CO2 concentration of this year. Defulat is 1980. THis function will pick the first value in the selected year,

Value

tacp The accumulation period for the annual max time series temperature

Author(s)

Arden Burrell, aburrell@whrc.org

`TSSRESTREND` *Time Series Segmentation of Residual Trends (MAIN FUNCTION)*

Description

Time Series Segmented Residual Trend (TSS.RESTREND) methodology.Takes in a complete monthly time series of a VI and its corrosponding precipitation (and temperature). It then looks looks for breakpoints using the BFAST function. The significance of the breakpoin in the residuals and the VPR is assessed using a Chow test, then, the total time series change is calculated.

Usage

```
TSSRESTREND(
  CTSR.VI,
  ACP.table = FALSE,
  ACT.table = NULL,
  CTSR.RF = FALSE,
  CTSR.TM = NULL,
  anu.VI = FALSE,
  acu.RF = FALSE,
  acu.TM = NULL,
```

```
  VI.index = FALSE,
  rf.b4 = FALSE,
  rf.af = FALSE,
  sig = 0.05,
  season = "none",
  exclude = 0,
  allow.negative = FALSE,
  allowneg.retest = FALSE,
  h = 0.15,
  retnonsig = FALSE
)
```

Arguments

`CTSR.VI`	Complete Monthly Time Series of Vegetation Index values. An object of class `'ts'` object without NA's.
`ACP.table`	A table of every combination of offset period and accumulation period.for precipitation ACP.table can be calculated using the `climate.accumulator`.
`ACT.table`	A table of every combination of offset period and accumulation period.for temperature ACP.table can be calculated using the `climate.accumulator`.
`CTSR.RF`	Complete Time Series of Rainfall. An object of class 'ts' object without NA's and be the same length and cover the same time range as CTSR.VI. If ACP.table is provided, CTSR.RF will be automitaclly calculated using the `ACP.calculator`
`CTSR.TM`	Complete Time Series of temperature. An object of class 'ts' object without NA's and be the same length and cover the same time range as CTSR.VI. Default (CTSR.TM=NULL). If ACT.table is provided, CTSR.RF will be automitaclly calculated using the `ACP.calculator`
`anu.VI`	The annual (Growing season) max VI. Must be a object of class `'ts'` without NA's. if anu.VI=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`acu.RF`	The optimal accumulated rainfall for anu.VI. Must be a object of class `'ts'` without NA's and be of equal length and temporal range to anu.VI. if anu.RF=FALSE, it will be calculated from ACP.table usingthe `AnnualClim.Cal`
`acu.TM`	The optimal accumulated rainfall for anu.TM. Must be a object of class `'ts'` without NA's and be of equal length and temporal range to anu.TM. if anu.TM=FALSE, it will be calculated from ACT.table usingthe `AnnualClim.Cal`
`VI.index`	the index of the CTSR.VI ts that the anu.VI values occur at. Must be the same length as anu.VI. NOTE. R indexs from 1 rather than 0. if VI.index=FALSE, it will be calculated from the CTSR.VI using `AnMaxVI`.
`rf.b4`	If a breakpoint in the VPR is detected this is the optimial accumulated rainfall before the breakpoint. must be the same length as the anu.VI. If ACP.table is provided it will be generated using `AnnualClim.Cal`
`rf.af`	If a breakpoint in the VPR is detected this is the optimial accumulated rainfall after the breakpoint. must be the same length as the anu.VI. If ACP.table is provided it will be generated using `AnnualClim.Cal`
`sig`	Significance of all the functions. defualt sig=0.05

`season` See `bfast`. This season value only applies to bfast done using the CTS VPR. if a non VPR adjusted BFAST is performed.a harmonic season is used.

`exclude` A numberic vector containg months excluded from breakpoint detection. This was included to allow sensor transitions to be masked.

`allow.negative` If true, will not preference positive slope in either CTSR or VI calculations. default=FALSE is set because negative associations between rainfall and vegetation in water limited ecosystems is unexpected If temperature data is included then this paramter is forced to TRUE.

`allowneg.retest` default=FALSE If temperature data is provided but found to not be significant then a retest is performed. This paramter is to allow negative on re-test.

`h` See `bfast`, The.minimal segment size between potentially detected breaks in the trend model given as fraction relative to the sample size (i.e. the minimal number of observations in each segment divided by the total length of the timeseries. Default h = 0.15.

`retnonsig` Bool. New in v0.3.0. Allows TSSRESTREND to return change estimates of values that filed the sig component in the residual analysis. defualt FALSE will give the same result as eralier versions.

Value

An object of class `'TSSRESTREND'` is a list with the following elements:

summary **Method** The method used to determine total change. (*RESTREND* see `RESTREND`, *segmented.RESTREND* see `seg.RESTREND`, *segmented.VPR* see `seg.VPR`)

- **Total.Change** The total significant change. Residual.Change + VPR.HeightChange.
- **Residual.Change** The change in the VPR Residuals over the time period
- **VPR.HeightChange** The change in VI at mean rainfall for a "ts" with a significant breakpoint in the VPR
- **model.p** p value of the regression model fitted to the VPR. See `lm`
- **residual.p** p value of the regression model fitted to the VPR Residuals. See `lm`
- **VPRbreak.p** the p value associated with the break height. See `lm`
- **bp.year** The Year of the most significant breakpoint

ts.data The Time series used in analysis. See Arguments for description

- CTSR.VI
- CTSR.RF
- anu.VI
- VI.index
- acu.RF
- StdVar.RF see `seg.VPR`)

ols.summary **chow.summary** summary of the most significant breakpoint.

- **chow.ind** Summary of every detected breakpoint
- **OLS.table** A matrix containing the coefficents for the CTS.fit, VPR.fit, RESTREND.fit and segVPR.fit

TSSRmodels models of class "lm" `lm` and class "bfast" `bfast` generated.

Note

if ACP.table = FALSE, CTSR.RF and acu.RF must be provided as well as rf.b4 and rf.af for 'ts' with a breakpoint in the VPR.

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

See Also

- plot.TSSRESTREND
- print.TSSRESTREND

Examples

```
## Not run:
#To get the latest version of the package (Still in development)
install.packages("devtools")
library("devtools")
install_github("ArdenB/TSSRESTREND", subdir="TSS.RESTREND")
library(TSS.RESTREND)
#Find the path of the rabbitRF.csv dataset, read it in and turn it into a time series
rf.path<- system.file("extdata", "rabbitRF.csv", package = "TSS.RESTREND", mustWork = TRUE)
in.RF <- read.csv(rf.path)
rf.data <- ts(in.RF, end=c(2013,12), frequency = 12)

#Find the path of the rabbitVI.csv dataset and read it in
vi.path <- system.file("extdata", "rabbitVI.csv", package = "TSS.RESTREND", mustWork = TRUE)
in.VI <- read.csv(vi.path)
CTSR.VI <- ts(in.VI, start=c(1982, 1), end=c(2013,12), frequency = 12)

#Define the max accumuulation period
max.acp <- 12
#Define the max offset period
max.osp <- 4
#Create a table of every possible precipitation value given the max.acp and max.osp
ACP.table <- climate.accumulator(CTSR.VI, rf.data, max.acp, max.osp)
results <- TSSRESTREND(CTSR.VI, ACP.table)
print(results)
plot(results, verbose=TRUE)

## End(Not run)
```

VPR.BFAST *BFAST Breakpoint Detector*

Description

takes the Complete VI and optimally accumulated Rainfall (and tmperature if included), calculates a `lm` between them And then performs a `bfast`.in the residuals. If BFAST.raw=TRUE, it will perform bfast on the Complete VI ts

Usage

```
VPR.BFAST(
  CTSR.VI,
  CTSR.RF,
  CTSR.TM = NULL,
  season = "none",
  BFAST.raw = FALSE,
  h = 0.15
)
```

Arguments

`CTSR.VI`	Complete Monthly Time Series of Vegetation Index values. An object of class `'ts'` object without NA's.
`CTSR.RF`	Complete Time Series of Rainfall. An object of class 'ts' object without NA's and be the same length and cover the same time range as CTSR.VI. If ACP.table is provided, CTSR.RF will be automitaclly calculated using the `ACP.calculator`
`CTSR.TM`	Complete Time Series of temperature. An object of class 'ts' object without NA's and be the same length and cover the same time range as CTSR.VI. Default (CTSR.TM=NULL).
`season`	See `bfast`. This season value only applies to bfast done using the CTS VPR. if a non VPR adjusted BFAST is performed.a harmonic season is used.
`BFAST.raw`	Defualt = FALSE If TRUE will perform a BFAST (season="harmonic") on the CTSR.VI If FALSE will perform BFAST on the CTSR VPR residuals
`h`	See `bfast`, The.minimal segment size between potentially detected breaks in the trend model given as fraction relative to the sample size (i.e. the minimal number of observations in each segment divided by the total length of the timeseries. Default h = 0.15.

Value

List of objects:

bkps The index of the Breakpoints detected. If no breakpoints are detected, bkps = FASLE

BFAST.obj See `bfast`

CTS.lm the `lm` of CTSR.VI and CTSR.RF

BFAST.type the type of BFAST done (VPR residuals or on the VI timeseris itself)

Author(s)

Arden Burrell, arden.burrell@unsw.edu.au

Examples

```
## Not run:
VPRBFdem <- VPR.BFAST(segVPRCTSR$cts.NDVI, segVPRCTSR$cts.precip)
print(VPRBFdem)
## End(Not run)
```

Index