Package 'climetrics'

April 18, 2024

Type Package

Title Climate Change Metrics

Version 1.0-15

Date 2024-04-18

Depends methods, R (>= 3.5.0), rts

Imports raster, terra, zoo, yaImpute, xts

Suggests R.rsp, tinytest

VignetteBuilder R.rsp

Description A framework that facilitates spatio-temporal analysis of climate dynamics through exploring and measuring different dimensions of climate change in space and time.

License GPL (>= 3)

Encoding UTF-8

URL https://r-gis.net/

BugReports https://github.com/shirintaheri/climetrics/issues/

NeedsCompilation no

Author Shirin Taheri [cre, aut] (<https://orcid.org/0000-0002-0303-3145>),
Babak Naimi [aut] (<https://orcid.org/0000-0001-5431-2729>),
Miguel Araujo [aut]

Maintainer Shirin Taheri <taheri.shi@gmail.com>

Repository CRAN

Date/Publication 2024-04-18 21:12:38 UTC

R topics documented:

---

aaClimate   *Changes in the Area of Analogous Climate*

Description

To quantify the change in area of analogous climates (classes),the change in area occupied by a given class between time 1 (t1) and time 2 (t2) periods is quantified.

Usage

```
aaClimate(precip,tmin, tmax, tmean,t1,t2)
aaClimateC(c1,c2)
```

Arguments

precip	A time series of precipitation as a Raster or Raster Time Series object
tmin	A time series of minimum temperature as a Raster or Raster Time Series object
tmax	A time series of maximum temperature as a Raster or Raster Time Series object
tmean	A time series of mean temperature as a Raster or Raster Time Series object; if not provided, it will be calculated from tmin and tmax
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2
c1	A single layer Raster layer contains Climate classes (e.g., Koppen Geiger climate classification) for time 1
c2	A single layer Raster layer contains Climate classes (e.g., Koppen Geiger climate classification) for time 2

Details

For a given cell with a given climate class, the change in area of analogous climates represented the ratio (in percentage) of the difference between time 2 and time 1 area of that class to the time 1 area of the same class. Positive values indicated gains in area, negative values indicated losses, and null values reflect no change.

The aaClimate first uses apply.months function to generate the monthly mean of climate parameter (results a Raster object with 12 layers correspond to 12 months) over each time period (t1 and t2); then uses the kgc function to generate the Koppen Geiger climate classification for each time period. Then the changes in the area of each class is calculated between time 1 and time 2. If the climate classification (regions) are available for time 1 and time 2, then the aaClimateC can be used instead.

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

`<taheri.shi@gmail.com>`; `<naimi.b@gmail.com>`

Examples

```
#-------
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmin <- rast(paste0(filePath,'/tmin.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))
tmean <- rast(paste0(filePath,'/tmean.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

class(n)
length(n)

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmin.t <- rts(tmin,n)
tmax.t <- rts(tmax,n)
tmean.t <- rts(tmean,n)
#------

pr.t # see the summary report of the raster time series object


###########################
# test of the metric:
#---------
#---------
# t1 (time1) = '1991/1990' takes all layers correspond to years between 1991-01-01 to 2000-12-31
# t2 (time2) = '2010/2020' takes all layers correspond to years between 2010-01-01 to 2020-12-31

aa <- aaClimate(precip=pr.t,tmin=tmin.t,tmax=tmax.t,tmean=tmean.t,t1='1991/2000',t2='2010/2020')
```

```
plot(aa,main="Changes in the area of Analogous Climates")


########
# Alternatively, if the climate in both times are available as climate classes
# (e.g., Koppen Geiger climate classification), changes in the area of analogous
# climate can be quantified using the aaClimateC function:

# Here, we first generate Koppen Geiger climate classification for time 1 and time 2, separately,
# To do so, we need the monthly average of climate variables (over years for each month):

# take the average of climate variables for each month over different years:

p12.1 <- apply.months(pr.t[['1991/2000']],'mean')
p12.2 <- apply.months(pr.t[['2010/2020']],'mean')

p12.1 # has 12 layers corresponding to 12 months

plot(p12.1)
#--
tmin12.1 <- apply.months(tmin.t[['1991/2000']],'mean')
tmin12.2 <- apply.months(tmin.t[['2010/2020']],'mean')
#--
tmax12.1 <- apply.months(tmax.t[['1991/2000']],'mean')
tmax12.2 <- apply.months(tmax.t[['2010/2020']],'mean')
#--
tmean12.1 <- apply.months(tmean.t[['1991/2000']],'mean')
tmean12.2 <- apply.months(tmean.t[['2010/2020']],'mean')
#--


##-------- now, the kgc function can be used to generate the climate classification map:
k1 <- kgc(p=p12.1,tmin = tmin12.1,tmax=tmax12.1, tmean = tmean12.1)
k2 <- kgc(p=p12.2,tmin = tmin12.2,tmax=tmax12.2, tmean = tmean12.2)

plot(k1, main= "Koppen Geiger climate classification - 1995")
plot(k2, main= "Koppen Geiger climate classification - 2015")

# Now, given the k1 and k2 climate classes, the changes in area of each class between two times
# are calculated using the aaClimateC function:

aa <- aaClimateC(k1,k2)

plot(aa)
```

---

apply.months   *Apply a function on rasters of each month over years*

Description

Apply a user specified function over raster time series for each month separately.

Usage

```
apply.months(x,FUN,...)
```

Arguments

x	A Raster or Raster Time Series object
FUN	A function to apply on rasters of each month over different years; The defaults if mean
...	additional argument: if x is a Raster or SpatRaster object, corresponding dates should be specified in the dates argument

Details

The output contains 12 raster layers corresponding to the 12 months (Jan-Dec.). For each of the available 12 months (from January to December), the corresponding rasters are selected and the FUN is applied. For example if the FUN is mean, the first layer in the output would be the mean of the rasters in January over different years, and the last later (12th layer) would be the mean of the rasters in December over different years.

Value

A Raster object (RasterStackBrick or SpatRaster depending on the input) contains 12 layers correspond to 12 months (From Jan to Dec)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder


pr <- rast(paste0(filePath,'/precip.tif'))

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresoinding dates

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
```

```
###########################

p12 <- apply.months(pr.t,'mean')


p12

plot(p12)
```

---

ccm   *Climate Change Metrics*

Description

A function to quantify a single or multiple climate change metrics between two time periods given the climatic raster time series variables.

Usage

```
ccm(x,...,stat,t1,t2,extreme,longlat,ny,names,verbose=TRUE)
```

Arguments

x	a Raster object or a Raster Time Series of a climate variable
...	additional climate variables can be included as a Raster object or a Raster Time Series
stat	a character specifying the metric used to generate climate change metrics, can be one or multiple metrics including 'sed','novelClimate' ,'localExtreme', 'aaClimate', 'daClimate', 'velocity', 'gVelocity', 'dVelocity' (or their abbreviations)
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2
extreme	the percentile used for stat='lce', local climate extreme event change
longlat	TRUE or FALSE, specifies whether the input data is geographic or projected
ny	integer; specifies the number of years between time 1 and time2, if the input is Raster time series, it will be identified by the function
names	character with a length equal to the number of input climate variables; specifies the names of the input climate variables; for some metrics the names (precipitation, tmin, tmax, tmean) is required
verbose	logical (default=TRUE), specifies whether the function print messages as a text output on the screen

Details

multiple climate change metrics are available to quantify multiple dimensions of climate change between two times.

The metrics include:

- sed: Standardized Local Anomalies

- localExtreme (lce): Changes in probability of local extremes

- aaClimate (aac): Changes in areas of analogous climates

- novelClimate (nc): Novel climates

- daClimate (dac): Changes in distance to analogous climates

- velocity (ve): Climate change velocity (based on the method represented in the paper Hamnan et al. (2015))

- dVelocity (dVe): Distance-based climate change velocity (based on the method used in Garcia et al. (2014))

- gVelocity (gVe): Gradient-based climate change velocity (based on the method represented in Burrow et al. (2011))

References:

- Burrows MT, Schoeman DS, Buckley LB, Moore P, Poloczanska ES, Brander KM, Brown C, Bruno JF, Duarte CM, Halpern BS, Holding J. (2011) The pace of shifting climate in marine and terrestrial ecosystems. Science. Nov 4;334(6056):652-5.

- Garcia, R.A., Cabeza, M., Rahbek, C. and Araújo, M.B. (2014). Multiple dimensions of climate change and their implications for biodiversity. Science, 344(6183).

- Hamann, A., Roberts, D.R., Barber, Q.E., Carroll, C. and Nielsen, S.E. (2015). Velocity of climate change algorithms for guiding conservation and management. Global Change Biology, 21(2), pp.997-1004.

Value

A Raster object (RasterLayer or RasterStackBrick or SpatRaster depending on the input) with single or multiple layers correspond to the selected metrics (stat)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

https://r-gis.net/

Examples

```
#-------
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):
```

```
pr <- rast(paste0(filePath,'/precip.tif'))
tmin <- rast(paste0(filePath,'/tmin.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))
tmean <- rast(paste0(filePath,'/tmean.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020


# n <- names(pr)
# n <- substring(n,2,11)
# head(n)
# n <- as.Date(n,format = '%Y.%m.%d')


n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

class(n)
length(n)

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmin.t <- rts(tmin,n)
tmax.t <- rts(tmax,n)
tmean.t <- rts(tmean,n)
#------

pr.t # see the summary report of the raster time series object

##########################
# test of the metric:
#---------
#---------
# t1 (time1) = '1991/1990' takes all layers correspond to years between 1991-01-01 to 2000-12-31
# t2 (time2) = '2010/2020' takes all layers correspond to years between 2010-01-01 to 2020-12-31


sed <- ccm(pr.t,tmin.t,tmean.t,tmax.t,t1='1991/2000',t2='2010/2020',stat='sed')

plot(sed, main='Standardized Local Anomalies')


s2 <- ccm(pr.t,tmin.t,tmean.t,tmax.t,t1='1991/2000',t2='2010/2020',stat=c('nc','sed'))

plot(s2)



#ve <- ccm(pr.t,tmin.t,tmean.t,tmax.t,t1='1991/2000',t2='2010/2020',stat=c('gve','dve'))
```

```
#plot(ve)

# following, the extreme argument is needed for localExtreme (lce) metric:
s3 <- ccm(pr.t,tmin.t,t1='1991/2000',t2='2010/2020',stat=c('localExtreme','sed')
          ,extreme = c(0.05,0.95))

plot(s3)
```

---

daClimate   *Changes in distance to Analogous Climates*

Description

Quantifies the changes in distances to analogous climates between two time periods.

Usage

```
daClimate(precip,tmin, tmax, tmean,t1,t2)
daClimateC(c1,c2)
```

Arguments

precip	A time series of precipitation as a Raster or Raster Time Series object
tmin	A time series of minimum temperature as a Raster or Raster Time Series object
tmax	A time series of maximum temperature as a Raster or Raster Time Series object
tmean	A time series of mean temperature as a Raster or Raster Time Series object; if not provided, it will be calculated from tmin and tmax
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2
c1	A single layer Raster layer contains Climate classes (e.g., Koggen Geiger climate classification) for time 1
c2	A single layer Raster layer contains Climate classes (e.g., Koggen Geiger climate classification) for time 2

Details

For a given cell, the function quantifies the distances to all cells with analogous climates in the time 1 period, i.e., belonging to the same climate class of the given cell. It also quantifies the distances to all cells that experience analogous climates in the time 2. Then, for each cell, it calculates the median of the great-circle distances (in km) below the 10th percentile of the distribution of all values, for both time 1 and time 2 periods, and maps the change over time. Negative values indicated a temporal decrease in distance, whereas positive values indicated an increase.

The daClimate first uses apply.months function to generate the monthly mean of climate parameter (results a Raster object with 12 layers correspond to 12 months) over each time period (t1 and t2); then uses the kgc function to generate the Koppen Geiger climate classification for each time period. Then the changes in the area of each class is calculated between time 1 and time 2. If the climate classification (regions) are available for time 1 and time 2, then the daClimateC can be used instead.

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
#-------
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmin <- rast(paste0(filePath,'/tmin.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))
tmean <- rast(paste0(filePath,'/tmean.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

class(n)
length(n)

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
```

```
tmin.t <- rts(tmin,n)
tmax.t <- rts(tmax,n)
tmean.t <- rts(tmean,n)
#------

pr.t # see the summary report of the raster time series object


############################
# test of the metric:
#---------
#---------
# t1 (time1) = '1991/1990' takes all layers correspond to years between 1991-01-01 to 2000-12-31
# t2 (time2) = '2010/2020' takes all layers correspond to years between 2010-01-01 to 2020-12-31

da <- daClimate(precip=pr.t,tmin=tmin.t,tmax=tmax.t,tmean=tmean.t,t1='1991/2000',t2='2010/2020')

plot(da,main="Changes in distance to Analogous Climates")


########
# Alternatively, if the climate in both times are available as climate classes
# (e.g., Koppen Geiger climate classification), changes in distance to analogous
# climate can be quantified using the daClimateC function:

# Here, we first generate Koppen Geiger climate classification for time 1 and time 2, separately,
# To do so, we need the monthly average of climate variables (over years for each month):

# take the average of climate variables for each month over different years:

p12.1 <- apply.months(pr.t[['1991/2000']],'mean')
p12.2 <- apply.months(pr.t[['2010/2020']],'mean')

p12.1 # has 12 layers corresponding to 12 months

plot(p12.1)
#--
tmin12.1 <- apply.months(tmin.t[['1991/2000']],'mean')
tmin12.2 <- apply.months(tmin.t[['2010/2020']],'mean')
#--
tmax12.1 <- apply.months(tmax.t[['1991/2000']],'mean')
tmax12.2 <- apply.months(tmax.t[['2010/2020']],'mean')
#--
tmean12.1 <- apply.months(tmean.t[['1991/2000']],'mean')
tmean12.2 <- apply.months(tmean.t[['2010/2020']],'mean')
#--

##-------- now, the kgc function can be used to generate the climate classification map:
k1 <- kgc(p=p12.1,tmin = tmin12.1,tmax=tmax12.1, tmean = tmean12.1)
k2 <- kgc(p=p12.2,tmin = tmin12.2,tmax=tmax12.2, tmean = tmean12.2)

plot(k1, main= "Koppen Geiger climate classification - 1995")
plot(k2, main= "Koppen Geiger climate classification - 2015")
```

```
# Now, given the k1 and k2 climate classes, the changes in distance to Analogous climate classes
# between two times is calculated using the daClimateC function:

da <- daClimateC(k1,k2)

plot(da)
```

---

dVelocity   *Distant-based Velocity of Climate Change*

Description

The method is developed based on the approach used in Garcia et al. (2014) Gradiant-based velocity is also implemented in the package and is available through the gVelocity function.

Garcia, R.A., Cabeza, M., Rahbek, C. and Araújo, M.B. (2014). Multiple dimensions of climate change and their implications for biodiversity. Science, 344(6183).

Usage

```
dVelocity(x,...,t1, t2, ny)
```

Arguments

x	The first input climate variable that can be a RasterStack or RasterBrick, or a raster time series
...	additional input climate variables in case of using multiple climate variables that can be entered as v1, v2, ...
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2
ny	integer; specifies the number of years between time 1 and time2, if the input is Raster time series, it is not needed as it will be identified by the function

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

pr <- rast(paste0(filePath,'/precip.tif'))
tmean <- rast(paste0(filePath,'/tmean.tif'))

n <- readRDS(paste0(filePath,'/dates.rds')) # corresoinding dates

head(n) # Dates corresponds to the layers in climate variables

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmean.t <- rts(tmean,n)
###########################


dv <- dVelocity(pr.t,tmean.t,t1='1991/2000',t2='2010/2020')

plot(dv)
```

---

gVelocity   *Gradient-based Velocity of Climate Change*

Description

The method is developed based on Burrow et al. (2011) which is a gradient-based velocity of climate change. Climate velocity can inform conservation planning and action in a warming world.

Distance-based velocity is also implemented in the package and is available through the ccm function.

Burrows MT, Schoeman DS, Buckley LB, Moore P, Poloczanska ES, Brander KM, Brown C, Bruno JF, Duarte CM, Halpern BS, Holding J. The pace of shifting climate in marine and terrestrial ecosystems. Science. 2011 Nov 4;334(6056):652-5..

Usage

```
gVelocity(x,...)
```

Arguments

x   a Raster object or a Raster Time Series of climate variable

...   additional arguments; not implemented

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))

#gv <- gVelocity(pr)

#plot(gv, main='Gradiant-based Velocity for Precipitation')
```

| kgc | *Koggen Geiger climate classification* |

Description

Given the monthly mean (12 months) of Precipitation, minimum, maximum, and mean temperature datasets, this function identify climate zones following Koppen Geiger climate classification rules.

Usage

```
kgc(p,tmin, tmax, tmean)
```

Arguments

p	A Raster object contains 12 months precipitation
tmin	A Raster object contains 12 months minimum temperature
tmax	A Raster object contains 12 months maximum temperature
tmean	A Raster object contains 12 months mean temperature

Details

If the input object is not a 12 month Raster object but a time series, the function first calls the apply.months to get the 12 month raster object.

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmin <- rast(paste0(filePath,'/tmin.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))
tmean <- rast(paste0(filePath,'/tmean.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

class(n)
length(n)

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmin.t <- rts(tmin,n)
tmax.t <- rts(tmax,n)
tmean.t <- rts(tmean,n)
#------

pr.t # see the summary report of the raster time series object


############################

# To generate Koppen Geiger climate classification, we need the monthly average of
# climate variables (over years for each month):

# you can use apply.months function to take the average of climate variables for each month
# over different years that generates 12 layers corresponds to 12 months:

p12 <- apply.months(pr.t[['1991/2000']],'mean')
```

```
p12

# plot(p12)

#--
tmin12 <- apply.months(tmin.t[['1991/2000']],'mean')

tmax12 <- apply.months(tmax.t[['1991/2000']],'mean')

tmean12 <- apply.months(tmean.t[['1991/2000']],'mean')


##-------- now, the kgc function can be used to generate the climate classification map:
k1 <- kgc(p=p12,tmin = tmin12,tmax=tmax12, tmean = tmean12)

k1

plot(k1)
```

---

localExtreme   *Changes in probability of local extremes*

Description

Probability of changes in local extremes is calculated at each cell based on either one or two climate variables (e.g., precipitation and temperature). For each climate variable, the extreme value (percentile of the extreme in the variable distribution) should be specified by user in the extreme parameter. For instance, if temperature and precipitation are specified as the first and second inputs (x1 and x2), extreme=c(0.95,0.05) may be used that considered 95th and 5th percentiles of the distribution of the temperature and precipitation time series, respectively over the first time period, and then it calculates the probability that the time1 percentile will be exceeded in time2. If two climate variables are used (e.g., temperature and precipitation), for each cell, the two probabilities are summed from which the product of the two probabilities are subtracted to obtain a measure of the probability of occurrence of either of the two extreme events (to avoid counting probabilities twice). The probability of time_1 extreme climates in time_2, is then subtracted from the probability of time_1 extreme climates to obtain the changes in the probability of extreme climate variables. Positive values indicated increased probability in time 2, whereas negative values indicated a decrease.

Usage

```
localExtreme(x1,x2,t1,t2,extreme)
```

Arguments

x1	Time Series of the first climate variable as a Raster or Raster Time Series Object
x2	Time Series of the second climate variable as a Raster or Raster Time Series Object; x1 and x2 can be lists of time series of climate variables in time 1 and time 2, respectively
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2
extreme	a numeric vector with a length of one or two (depends on whether one or two climate variables are used as input), specifying the percentile of extreme value in the input (first value corresponds to x1 and second corresponds to x2 climate variables)

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmax.t <- rts(tmax,n)

###########################
# test of the metric:
```

```
# The extreme argument corresponds to the first and second climate variables
# (i.e., x1 and x2; precipitation and temperature) that specify the percentile of the extreme
# condition in climate variable; here, 0.05 is used for precipitation; and 0.95 for temperature

le <- localExtreme(x1=pr.t,x2=tmax.t,t1='1991/2000',t2='2010/2020', extreme = c(0.05, 0.95))

plot(le, main='Probability of Changes in Local Climate Extreme')
```

---

novelClimate   *Novel Climates*

Description

Novel climate is one of the climate change metrics to quantify the dissimilarities between climate
parameters between two time periods. This metric uses standard euclidean distance between each
cell in Time 2 and all cells in Time 1 and retains the minimum of those distances. The inter-annual
standard deviation for each parameter is used for the standardization. The larger the score, the more
dissimilar the climate in Time 2 is in relation to the global pool of potential climatic analogues.

Usage

```
novelClimate(x,...,t1,t2)
```

Arguments

x	a Raster object or a Raster Time Series of a climate variable
...	additional climate variables can be included as a Raster object or a Raster Time Series
t1	a character or a numeric vector, specifying the index of raster layers for time 1; if x is missing, t1 and t2 can be list of climate variables in time 1 and time 2, respectively.
t2	a character or a numeric vector, specifying the index of raster layers for time 2; if x is missing, t1 and t2 can be list of climate variables in time 1 and time 2, respectively.

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmean <- rast(paste0(filePath,'/tmean.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmean.t <- rts(tmean,n)

###########################
# test of the metric:

n1 <- novelClimate(pr.t,tmean.t,t1='1991/2000',t2='2010/2020')

plot(n1, main='Novel Climate')

######
# if the input object is SpatRaster (or RasterStack or RasterBrick) object:
# t1 and t2 would be the numbers specifying which layers correspond to time1 and time2:

n2 <- novelClimate(pr,tmean,t1=1:120,t2=229:360)

plot(n2)
```

---

sed   *Standardized Local Anomalies*

Description

Standardized Local Anomalies is one of the climate change metrics to quantify the change between two time periods.

Usage

```
sed(x,...,t1,t2)
```

Arguments

x	a Raster object or a Raster Time Series of a climate variable
...	additional climate variables can be included as a Raster object or a Raster Time Series
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmin <- rast(paste0(filePath,'/tmin.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresponding dates

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmin.t <- rts(tmin,n)
tmax.t <- rts(tmax,n)

###########################
# test of the metric:

s <- sed(pr.t,tmin.t,tmax.t,t1='1991/2000',t2='2010/2020')

plot(s, main='Standardized Local Anomalies')
```

```
######
# if the input object is SpatRaster (or RasterStack or RasterBrick) object:
# t1 and t2 would be the numbers specifying which layers correspond to time1 and time2:

s <- sed(pr,tmin,tmax,t1=1:120,t2=229:360)

plot(s)
```

---

temporalTrend   *Temporal Trend (Slope) of a climate variable*

Description

The method calculates the trend of a climate variable changes over time. The function use a Raster time series as input and returns a Raster object that represent the trend at the pixel level.

Usage

```
temporalTrend(x,...)
```

Arguments

x   a Raster object or a Raster Time Series of climate variable
...   additional arguments; not implemented

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))

tr <- temporalTrend(pr)
```

```
# plot(tr, main='Trend (slope) of Precipitation time series')
```

---

velocity   *Velocity of Climate Change*

Description

The method is developed based on the method represented in the paper Hamnan et al. (2015). Two additional functions including Distance-based (dVelocity) and Gradiant-based velocity (gVelocity) are also implemented and available in the package.

- Hamann, A., Roberts, D.R., Barber, Q.E., Carroll, C. and Nielsen, S.E. (2015). Velocity of climate change algorithms for guiding conservation and management. Global Change Biology, 21(2), pp.997-1004.

Usage

```
velocity(x1,x2,t1, t2,...)
```

Arguments

x1	The first input climate variable that can be a RasterStack or RasterBrick, or a raster time series
x2	The second input climate variable that can be a RasterStack or RasterBrick, or a raster time series
t1	a chanracter or a numeric vector, specifying the index of raster layers for time 1
t2	a chanracter or a numeric vector, specifying the index of raster layers for time 2
...	not implemented

Value

A single Raster layer (RasterLayer or SpatRaster depending on the input)

Author(s)

Shirin Taheri; Babak Naimi

<taheri.shi@gmail.com>; <naimi.b@gmail.com>

Examples

```
filePath <- system.file("external/", package="climetrics") # path to the dataset folder

# read the climate variables using the terra package (you can use the raster package as well):

pr <- rast(paste0(filePath,'/precip.tif'))
tmax <- rast(paste0(filePath,'/tmax.tif'))

pr # has 360 layers corresponds to months of the years 1991-2020

n <- readRDS(paste0(filePath,'/dates.rds')) # read corresoinding dates

head(n) # Dates corresponds to the layers in climate variables (pr, tmin, tmax, tmean)

####################

# use rts function in the rts package to make a raster time series:

pr.t <- rts(pr,n)
tmax.t <- rts(tmax,n)

###########################
# test of the metric:

# The extreme argument corresponds to the first and second climate variables
# (i.e., x1 and x2; precipitation and temperature) that specify the percentile of the extreme
# condition in climate variable; here, 0.05 is used for precipitation; and 0.95 for temperature

ve <- velocity(x1=pr.t,x2=tmax.t,t1='1991/2000',t2='2010/2020')

# plot(ve, main='Velocity of Climate Change')
```

Index