Package ‘RGENERATEPREC’

October 12, 2022

Maintainer  Emanuele Cordano <emanuele.cordano@gmail.com>
License     GPL (>= 2)
Title       Tools to Generate Daily-Precipitation Time Series
Type        Package
Author      Emanuele Cordano
Description The method ‘generate()’ is extended for spatial multi-site
stochastic generation of daily precipitation. It generates precipitation
occurrence in several sites using logit regression (Generalized Linear

Version  1.2.9
Repository CRAN
Date       2022-01-11
Depends   R (>= 3.5.0), copula, RGENERATE, blockmatrix, Matrix, stringr
Imports    RMAWGEN
Suggests  knitr,rmarkdown,lubridate,mapview,sf,lmom,ggplot2,reshape2,RefManageR
VignetteBuilder knitr
URL        https://github.com/ecor/RGENERATEPREC
RoxygenNote 7.1.2
NeedsCompilation no
Date/Publication 2022-01-20 16:12:49 UTC

R topics documented:

  CCGamma .................................................. 2
  CCGammaToBlockmatrix .................................... 5
  dw.spell ................................................... 6
  generate.PrecipitationOccurrenceModel .................. 8
  nwdays .................................................... 11
  omega ..................................................... 13
  omega_inv ............................................... 14
This function extends continuity_ratio and adds the corresponding gaussian correlation matrix for no-precipitation occurrence.

**Description**

This function extends continuity_ratio and adds the corresponding gaussian correlation matrix for no-precipitation occurrence.

**Usage**

```r
CCGamma(
  data,
  lag = 0,
  p0_v1 = NULL,
  p = NA,
  valmin = 0.5,
  nearPD = (lag >= 0),
  interval = c(-1, 1),
  tolerance = .Machine$double.eps,
  only.matrix = FALSE,
  return.value = NULL,
  null.gcorrelation = 1e-05,
  sample = NULL,
  origin = "1961-1-1",
  ...
)
```

**Arguments**

- `data`: data frame or 'zoo' R object containing daily precipitation time series for several gauges (one gauge time series per column). See continuity_ratio.
- `lag`: numeric lag (expressed as number of days) used for computation for "cross" continuity ratio and joint probability of precipitation (no)occurrence. See continuity_ratio.
- `p0_v1`: vector for marginal probabilities, see omega and omega_inv.
- `p`: positive integer parameter. Default is NA, otherwise, lag is calculated as the vector 0:p.
- `valmin`: threshold precipitation value [mm] for wet/dry day indicator. If precipitation is lower than valmin, day is considered dry. Default is 0.5 mm. See continuity_ratio.
nearPD see omega_inv. Default is (lag==0).
interval, tolerance see omega_inv
only.matrix logical value. If TRUE the function returns only the gaussian correlate matrix. Default is FALSE.
return.value string. If it is not either NULL (Default) and NA, function returns only the argument indicated by this argument.
null.gcorrelation numerical value nooccurrence_gcorrelation under which is considered to be 0.
sample character string indicated if function must be calculated differently for subset of the year, e.g. monthly. Admitted values are NULL (Default), "all" or "monthly".
origin character string (yyyy-dd-mm) indicated the date of the first row of "data". It is used if data and sample are not NULL.
... additional arguments of omega_inv or CCGamma

Value
An object which is a list containing the following fields:
continuity_ratio: lag-day lagged continuity ratio, as returned by continuity_ratio;
occurrence: joint probability of lag-day lagged precipitation occurrence, as returned by continuity_ratio;
nooccurrence: joint probability of lag-day lagged no precipitation occurrence, as returned by continuity_ratio;
lag: number of days lagged between the two compared events (see argument lag);
p0_v1: vector of marginal probability of no precipitation occurrence. If lag is 0, it corresponds to the diagonal of nooccurrence matrix (see argument p0_v1);
nooccurrence_gcorrelation corresponding gaussian correlation for no precipitation occurrence obtained by applying omega_inv to nooccurrence,

If the argument only.matrix is TRUE, only nooccurrence_gcorrelation is returned as a matrix.
In case the argument lag is a vector with length more than one, the function returns a list of the above-cited return object for each value of the vector lag.

Note
This function is useful to generate the serial cross-correlation matrices for no precipitation occurrence for Yule-Walker Equations. In case lag is a vector, nearPD must be a vector of the same size, default is (lag==0).
See the R code for major details

Author(s)
Emanuele Cordano
References


See Also

continuity_ratio, omega_inv, omega, CCGammaToBlockmatrix

Examples

data(trentino)

year_min <- 1961
year_max <- 1990
origin <- paste(year_min,1,1,sep="-")

period <- PRECIPITATION$year>year_min & PRECIPITATION$year<=year_max
station <- names(PRECIPITATION)[!(names(PRECIPITATION) %in% c("day","month","year"))]
prec_mes <- PRECIPITATION[period,station]

# removing nonworking stations (e.g. time series with NA)
accepted <- array(TRUE,length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it]))
}
prec_mes <- prec_mes[,accepted]
# the dataset is reduced!!!
prec_mes <- prec_mes[,1:2]

CCGamma <- CCGamma(data=prec_mes,lag=0,tolerance=0.001,only.matrix=FALSE)

# Not Run in the examples, uncomment to run the following line
CCGamma <- CCGamma(data=prec_mes,lag=0:2,tolerance=0.001,only.matrix=FALSE)

# Not Run in the examples, uncomment to run the following line
CCGamma_monthly <- CCGamma(data=prec_mes,lag=0,tolerance=0.001,only.matrix=FALSE, sample="monthly",origin=origin)
This function returns a blockmatrix object containing the gaussian cross-correlation matrices.

Usage

CCGammaToBlockmatrix(data, lag = 0, p = 3, ...)

Arguments

data data frame or 'zoo' R object containing daily precipitation time series for several gauges (one gauge time series per column). See CCGamma.

lag numeric (expressed as number of days) used for the element [1,1] of the returned blockmatrix.

p numeric order $p$ of the auto-regression

... further arguments of CCGamma

Details

This a wrapper for CCGamma with the option only.matrix=TRUE and the function value is transformed into a blockmatrix object.

Value

A blockmatrix object containing the gaussian cross-correlation matrices.

See Also

CCGamma, continuity_ratio, omega_inv, omega

Examples

data (trentino)

year_min <- 1961
year_max <- 1990

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
station <- names(PRECIPITATION)[!(names(PRECIPITATION) %in% c("day","month","year")]
prec_mes <- PRECIPITATION[period,station]

## removing nonworking stations (e.g. time series with NA)
accepted <- array(TRUE,length(names(prec_mes)))
It calculates dry/wet spell duration.

Usage

dw.spell(}
dw.spell

data, 
valmin = 0.5, 
origin = "1961-1-1", 
extract = NULL, 
month = 1:12, 
melting.df = FALSE, 
from.start = FALSE, 
only.inner = FALSE
)

Arguments

data data frame R object containing daily precipitation time series for several gauges (one gauge time series per column).
valmin threshold precipitation value [mm] for wet/dry day indicator.
origin character string "yyyy-mm-dd" indicated the date of the first row of "data".
extract string character referred to the state to be extracted, eg. "dry" or "wet"
month integer vectors containing the considered months. Default is 1:12 (all the year).
melting.df logical value. If it TRUE the output is melted into a data frame. Default is FALSE.
from.start logical value. If is TRUE the spell is referenced to its first day, if it is FALSE (default) the spell is referenced to its last date.
only.inner logical value. It is used in case extract is not NULL, if the value is TRUE, it extracts dry/wet spells completely inside the selected month period. Default is FALSE.

Value

Function returns a list of data frames containing the spell length expressed in days

Examples

data(trentino)

year_min <- 1961
year_max <- 1990

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
station <- names(PRECIPITATION)[!(names(PRECIPITATION) %in% c("day","month","year"))]
prec_mes <- PRECIPITATION[period,station]

## removing nonworking stations (e.g. time series with NA)
accepted <- array(TRUE,length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it]))
}

prec_mes <- prec_mes[,accepted]
## the dataset is reduced!!
prec_mes <- prec_mes[,1:3]

origin <- paste(year_min,1,1,sep="-")
dw_spell <- dw.spell(prec_mes,origin=origin)
dw_spell_dry <- dw.spell(prec_mes,origin=origin,extract="dry")

hist(dw_spell_dry$T0001$spell_length)

## Single Gauging Station

prec_mes <- prec_mes[,1]

origin <- paste(year_min,1,1,sep="-")
dw_spell <- dw.spell(prec_mes,origin=origin)
dw_spell_dry <- dw.spell(prec_mes,origin=origin,extract="dry")
dw_spell_dry_start <- dw.spell(prec_mes,origin=origin,extract="dry",
month=5:8,from.start=TRUE) ## dry spell

dw_spell_dry_start_2 <- dw.spell(prec_mes,origin=origin,extract="dry",
month=5:8,from.start=TRUE,only.inner=TRUE) ## dry spell
## is referenced to the first day instead of the latest one as default.

hist(dw_spell_dry[[1]]$spell_length)

generate.PrecipitationOccurrenceModel

Stochastic Generation of a PrecipitationOccurrenceModel or PrecipitationOccurrenceMultiSiteModel model object

Description

It is an implementation of generate method

Usage

## S3 method for class 'PrecipitationOccurrenceModel'
generate(
  x,
  newdata = NULL,
  previous = NULL,
  n = 30,
  random = runif(n, min = 0, max = 1),
  exogen = NULL,
  monthly.factor = NULL,
generate.PrecipitationOccurrenceModel

...)

## S3 method for class 'CCGammaObjectListPerEachMonth'
genenerate(x, ...)

## S3 method for class 'PrecipitationOccurrenceMultiSiteModel'
genenerate(
  x,
  exogen,
  n = 10,
  origin = "1961-1-1",
  end = "1990-1-1",
  previous = NULL,
  monthly.factor = NULL,
  ...
)

## S3 method for class 'PrecipitationAmountModel'
genenerate(x, ...)

Arguments

x model returned by PrecipitationOccurrenceModel or PrecipitationOccurrenceMultiSiteModel
newdata predictor or exogenous variables. See predict.PrecipitationOccurrenceModel
previous logical vector containing previously occurred states
n number of generations. See generate. Here it is ignored and the number of
generations is given by origin, end or monthly.factor.
random vector of random or calculated numbers ranging between 0 and 1
exogen predictor or exogenous variables
monthly.factor vector of factors indicating the month of the days
... further arguments
origin, end character strings (yyyy-dd-mm) indicating the start and/or end date of the daily weather generation.

Value

A vector or a data frame reporting generated time series for each station.

References


See Also

generate.predict.glm, PrecipitationOccurrenceModel, PrecipitationOccurrenceMultiSiteModel

Examples

library(RGENERATEPREC)

## A function example can be found in the following script file:
scriptfile <- system.file("example.generate.R", package="RGENERATEPREC")
## The current file path is given by 'scriptfile' variable:
print(scriptfile)
## To run the example file, launch the file with 'source' command (uncomment the following line)
#source(scriptfile)
## ALTERNATIVELY you can run the following lines:

data(trentino)

year_min <- 1961
year_max <- 1990

origin <- paste(year_min, 1, 1, sep="-")
end <- paste(year_max, 12, 31, sep="-")

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
period_temp <- TEMPERATURE_MAX$year>=year_min & TEMPERATURE_MAX$year<=year_max

prec_mes <- PRECIPITATION[period,]
Tx_mes <- TEMPERATURE_MAX[period_temp,]
Tn_mes <- TEMPERATURE_MIN[period_temp,]
accepted <- array(TRUE, length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  acc <- TRUE
  acc <- (length(which(!is.na(Tx_mes[, it])))==length(Tx_mes[, it])) & acc
  acc <- (length(which(!is.na(Tn_mes[, it])))==length(Tn_mes[, it])) & acc
  accepted[it] <- (length(which(!is.na(prec_mes[, it])))==length(prec_mes[, it])) & acc
}

valmin <- 1.0
prec_mes <- prec_mes[, accepted]

Tx_mes <- Tx_mes[, accepted]
Tn_mes <- Tn_mes[, accepted]
prec_occurrence_mes <- prec_mes>=valmin
nwetdays

It calculates the number of wet days for each month and each year

**Description**

It calculates the number of wet days for each month and each year
nwetdays

Usage

nwetdays(data, valmin = 0.5, origin = "1961-1-1", station = names(data))

Arguments

data  data frame R object containing daily precipitation time series for several gauges (one gauge time series per column).
valmin threshold precipitation value [mm] for wet/dry day indicator.
origin character string "yyyy-mm-dd" indicated the date of the first row of "data".
station character string indicating the stations. Default is names(data)

Value

Function returns a list of data frames containing the spell length expressed in days

Examples

data(trentino)

year_min <- 1961
year_max <- 1990

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
station <- names(PRECIPITATION)[!(names(PRECIPITATION) %in% c("day","month","year"))]
prec_mes <- PRECIPITATION[period,station]

## removing nonworking stations (e.g. time series with NA)
accepted <- array(TRUE,length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it]))
}

prec_mes <- prec_mes[,accepted]
## the dataset is reduced!!
prec_mes <- prec_mes[,1:3]

origin <- paste(year_min,1,1,sep="-")
nwetdays <- nwetdays(prec_mes,origin)
### Description

This function finds the bivariate joint probability or the binary correlation from the corresponding Gaussian correlation $x$.

### Usage

```r
omega(x = 0.5, p0_v1 = 0.5, p0_v2 = NA, correlation = FALSE)
```

### Arguments

- **x**: value of expected correlation between the corresponding Gaussian-distributed variables.
- **p0_v1, p0_v2**: probability of no precipitation occurrences for the v1 and v2 time series respectively. See Notes.
- **correlation**: logical numeric value. Default is FALSE. If TRUE the function returns the binary correlation like eq. 6 of Mhanna, et al., 2011.

### Value

Probability of no precipitation occurrence in both v1 and v2 simultaneously. It is a matrix if $x$ is a matrix.

### Note

This function makes use of normal copula. A graphical introduction to this function (with its inverse) makes is present in the following URL references: [https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/joc.2305](https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/joc.2305) and [https://www.sciencedirect.com/science/article/pii/S0022169498001863](https://www.sciencedirect.com/science/article/pii/S0022169498001863) (See fig. 1 and par. 3.2). If the argument $p0_v2$, the two marginal probability values must be given as a vector through the argument $p0_v1$: $p0_v1=c(p0_v1,p0_v2)$. In case $x$ is a correlation/covariance matrix the marginal probabilities are given as a vector through the argument $p0_v1$.

### Author(s)

Emanuele Cordano

### References


See Also

normalCopula, pcopula

Examples

rho <- 0.4
p00 <- omega(x=rho,p0_v1=0.5,p0_v2=0.5)
cor00 <- omega(x=rho,p0_v1=0.5,p0_v2=0.5,correlation=TRUE)

omega_inv

This function is the inverse of omega function

Description

This function is the inverse of omega function

Usage

omega_inv(
  p0 = NULL,
  p0_v1 = 0.5,
  p0_v2 = p0_v1,
  p00 = p0_v1 * p0_v2,
  correlation = NA,
  only.value = TRUE,
  interval = c(-1,1),
  tolerance = 0.001,
  nearPD = TRUE,
  force.independence = TRUE,
  ...
)

Arguments

p0

matrix of joint probabilities. Default is NULL, otherwise functions returns a matrix with values

p0_v1, p0_v2

probability of no precipitatin occurrences for the v1 and v2 time series respectively.

p00

probability of no precipitation occurrence in both v1 and v2 simultaneously returned by omega
correlation  numerical value. Default is NA. Binary correlation returned by `omega` when the argument `correlation=TRUE` (see `omega_root`).

only.value logical value. If `TRUE` (Default) the only Gaussian correlation (x input variable of `omega`) is returned, otherwise the complete output of `uniroot` is returned.

interval see interval option of `uniroot`. Default is `c(-1,1)`.

tolerance tolerance (numeric) parameter used for comparisons with the extreme value of marginal probabilities. Default is 0.001.

nearPD logical. If `TRUE` (Default) a positive-definite correlation matrix is returned by applying `nearPD` in case `p0` is a matrix and not NULL.

force.independence logical value. Default is `TRUE`. If it is `TRUE`, no negative correlation is considered and negative values of correlation are forced to be 0 (independence).

... further arguments for `uniroot`

Value

table of expected correlation between the corresponding Gaussian-distributed variables (see x input argument of `omega`.

Note

This function finds the zero of the `omega_root` function by calling `uniroot`. If the argument `p0` is not NULL and is a matrix of joint probabilities, the function returns a correlation matrix by using the elements of `p0` as joint probabilities for each couple and `p0_v1` as a vector of marginal probability of each occurrence/no-occurrence (In this case if the length of `p0_v1` does not correspond to the number of columns of `p0`, the marginal probabilities are taken from the diagonal of `p0`). See the R code for major details.

Author(s)

Emanuele Cordano

See Also

`normalCopula`, `pcopula`, `omega` (and reference URLs therein)

Examples

```r
x <- omega_inv(p0_v1=0.5,p0_v2=0.5,p0=1.1*0.5*0.5)
omega(x,p0_v1=0.5,p0_v2=0.5)
```
omega_root

This is the target function whose zero is searched to create the inverse function of omega.

Description

This is the target function whose zero is searched to create the inverse function of omega.

Usage

omega_root(
  x = 0.5,
  p0_v1 = 0.5,
  p0_v2 = 0.5,
  p00 = p0_v1 * p0_v2,
  correlation = NA
)

Arguments

x
  value of expected correlation between the corresponding Gaussian-distributed variables

p0_v1, p0_v2
  probability of no precipitation occurrences for the v1 and v2 time series respectively.

p00
  probability of no precipitation occurrence in both v1 and v2 simultaneously returned by omega

correlation
  numerical value. Default is NA. Binary correlation retured by omega when the argument correlation=TRUE

Value

the value p00-omega(x=x,p0_v1=p0_v1,p0_v2=p0_v2) or correlation-omega(x=x,p0_v1=p0_v1,p0_v2=p0_v2) (if correlation is not NA)

Note

This function makes use of normal copula

Author(s)

Emanuele Cordano

See Also

normalCopula, pcopula, omega, omega_inv
PrecipitationAmountModel

`PrecipitationAmountModel` creates a Precipitation Amount Model

### Description

Creates a Precipitation Amount Model

### Usage

```r
PrecipitationAmountModel(
  x,
  valmin = 1,
  station = names(x),
  sample = "monthly",
  origin = "1961-1-1",
  ...
)
```

### Arguments

- **x**: observed precipitation amount time series (data frame)
- **valmin**: maximum admitted value of precipitation depth
- **station**: string vector containing station identification codes
- **sample**: character string. If it is "monthly" (Default), the correlation matrix is calculated per each month.
- **origin**: date of the day referred by he first row of `x`.
- **...**: further arguments for `normalizeGaussian_severalstations`

### Value

The function returns an S3 object containing the correlation matrix of precipitation amount values (excluding the zeros). In case `sample` is "monthly", the function returns a `MonthlyList` S3 object.

### See Also

- `predict.PrecipitationAmountModel`
- `normalizeGaussian_severalstations`
- `generate`
Examples

```r
set.seed(1245)
data(trentino)

year_min <- 1961
year_max <- 1990

origin <- paste(year_min,1,1,sep="-")
end <- paste(year_max,12,31,sep="-"

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
period_temp <- TEMPERATURE_MAX$year>=year_min & TEMPERATURE_MAX$year<=year_max

prec_mes <- PRECIPITATION[period,]
Tx_mes <- TEMPERATURE_MAX[period_temp,]
Tn_mes <- TEMPERATURE_MIN[period_temp,]
accepted <- array(TRUE,length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  acc <- TRUE
  acc <- (length(which(!is.na(Tx_mes[,it])))==length(Tx_mes[,it])) & acc
  accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it])) & acc
}

valmin <- 1.0
prec_mes <- prec_mes[,accepted]

Tx_mes <- Tx_mes[,accepted]
Tn_mes <- Tn_mes[,accepted]
prec_occurrence_mes <- prec_mes>=valmin
station <- names(prec_mes)[!(names(prec_mes) %in% c("day","month","year"))]
precamount <- PrecipitationAmountModel(prec_mes,station=station,origin=origin)
val <- predict(precamount)
prec_gen <- generate(precamount)

month <- adddate(as.data.frame(residuals(precamount$T0090)),origin=origin)$month
###plot(month,residuals(precamount$T0090))
plot(factor(month),residuals(precamount$T0090))
```
Precipitation Occurrence Model

Description

This function creates a stochastic Occurrence Model for the variable x (PrecipitationOccurrenceModel S3 object) through a calibration from observed data.
Usage

PrecipitationOccurrenceModel(
  x,
  exogen = NULL,
  p = 1,
  monthly.factor = NULL,
  valmin = 0.5,
  id.name = NULL,
  ...
)

Arguments

x variable utilized for the auto-regression of its occurrence, e.g. daily precipitation
exogen exogenous predictors
p auto-regression order
monthly.factor vector of factors indicating the month of the days
valmin minimum admitted value for daily precipitation amount
id.name identification name of the station
... further arguments

Value

The function returns a PrecipitationOccurrenceModel-class S3 object containing the following elements:
predictor data frame containing the endogenous and exogenous predictors of the logistic regression model;
glm the generalized linear model using for the logistic regression;
p auto-regression order
valmin minimum admitted value for daily precipitation amount

See Also

glm

Examples

library(RGENERATEPREC)
data(trentino)
year_min <- 1961
year_max <- 1990
period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
period_temp <- TEMPERATURE_MAX$year>=year_min & TEMPERATURE_MAX$year<=year_max

prec_mes <- PRECIPITATION[period_temp]  
Tx_mes <- TEMPERATURE_MAX[period_temp]  
Tn_mes <- TEMPERATURE_MIN[period_temp]  
accepted <- array(TRUE,length(names(prec_mes)))  
names(accepted) <- names(prec_mes)  
for (it in names(prec_mes)) {  
acc <- TRUE  
acc <- (length(which(!is.na(Tx_mes[,it])))==length(Tx_mes[,it])) & acc  
acc <- (length(which(!is.na(Tn_mes[,it])))==length(Tn_mes[,it])) & acc  
accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it])) & acc  
}

valmin <- 1.0  
prec_mes <- prec_mes[,accepted]

Tx_mes <- Tx_mes[,accepted]  
Tn_mes <- Tn_mes[,accepted]  
prec_occurrence_mes <- prec_mes>=valmin

station <- names(prec_mes)[!(names(prec_mes) %in% c("day","month","year"))]  
it <- station[2]  
vect <- Tx_mes[,it]-Tn_mes[,it]  
months <- factor(prec_mes$month)  
model <- PrecipitationOccurrenceModel(x=prec_mes[,it],exogen=vect,monthly.factor=months)  
probs <- predict(model$glm,type="response")

plot(months[-1],probs)  
newdata <- model$predictor[2000:2007,]  
probs0 <- predict(model,newdata=newdata)
Description

This functions creates a stochastic Occurrence Multi-Site Model for the variable \( x \) (PrecipitationOccurrenceMultiSiteModel S3 object) through a calibration from observed data.

Usage

```r
PrecipitationOccurrenceMultiSiteModel(
  x,
  exogen = NULL,
  station = names(x),
  origin = origin,
  valmin = 0.5,
  multisite_type = "wilks",
  tolerance_wilks = 0.001,
  p = 2,
  ...
)
```

Arguments

- **x**: data frame (each column is a site) of variable utilized for the auto-regression of its occurrence, e.g. daily precipitation
- **exogen**: exogenous predictors
- **station**: character string vectors containing the codes of the station used for model calibration
- **origin**: character string (yyyy-dd-mm) indicating the date of the first row of "x".
- **valmin**: minimum admitted value for daily precipitation amount
- **multisite_type**: string indicating the utilized approach for spatial multi-site dependence description. Default is "wilks".
- **tolerance_wilks**: see tolerance used by \( \omega_{inv} \) through \texttt{CCGamma}
- **p**: auto-regression order
- **...**: further arguments

Value

The function returns a \texttt{PrecipitationOccurrenceModel-class} S3 object containing the following elements:

- \texttt{PrecipitationOccurrenceModel} S3 class objects for each analyzed site. The name is the site (or station) code
- \texttt{ccgama} \texttt{CCGammaObjectListPerEachMonth} object, i.e. matrices of Gaussian Inter-Site Correlation returned by \texttt{CCGamma};
- **type**: string indicating the utilized approach for spatial multi-site dependence description, only "wilks" type is implemented;
- **station**: character string vectors containing the codes of the station used in PrecipitationMultiSiteOccurrenceModel.
See Also

PrecipitationOccurrenceModel, CCGamma

Examples

library(RGENERATEPREC)

data(trentino)

year_min <- 1961
year_max <- 1990
origin <- paste(year_min,1,1,sep="-")

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
period_temp <- TEMPERATURE_MAX$year>=year_min & TEMPERATURE_MAX$year<=year_max

prec_mes <- PRECIPITATION[period,]
Tx_mes <- TEMPERATURE_MAX[period_temp,]
Tn_mes <- TEMPERATURE_MIN[period_temp,]
accepted <- array(TRUE,length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  acc <- TRUE
  acc <- (length(which(!is.na(Tx_mes[,it])))==length(Tx_mes[,it])) & acc
  acc <- (length(which(!is.na(Tn_mes[,it])))==length(Tn_mes[,it])) & acc
  accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it])) & acc
}

valmin <- 1.0
prec_mes <- prec_mes[,accepted]

Tx_mes <- Tx_mes[,accepted]
Tn_mes <- Tn_mes[,accepted]
prec_occurrence_mes <- prec_mes>=valmin

station <- names(prec_mes)[!(names(prec_mes) %in% c("day","month","year"))]
station <- station[1:2] # to save example elapsed time!!
exogen <- Tx_mes-Tn_mes
months <- factor(prec_mes$month)

' ### Not Run!!
# The following lines are commented to save example elapsed time!!
model_multisite <- PrecipitationOccurrenceMultiSiteModel(x=prec_mes,exogen=exogen,
origin=origin,multisite_type="wilks")

### Not Run!!
# The following lines are commented to save example elapsed time!!
model_multisite_logit <- PrecipitationOccurrenceMultiSiteModel(x=prec_mes,exogen=exogen,
predict.PrecipitationOccurrenceModel

Prediction of a PrecipitationOccurrenceModel model object

Description

It is a wrapper of predict.glm method for the a PrecipitationOccurrenceModel model object S3 class.

Usage

## S3 method for class 'PrecipitationOccurrenceModel'
predict(
  object,
  newdata = NULL,
  type = "response",
  previous = NULL,
  endogenous = NULL,
  ...
)

## S3 method for class 'PrecipitationOccurrenceMultiSiteModel'
predict(object, ...)

## S3 method for class 'PrecipitationAmountModel'
predict(
  object,
  newdata = NULL,
  origin_newdata = NA,
  precipitation.value.random.generation = FALSE,
  ...
)

Arguments

- **object**: model returned by PrecipitationOccurrenceModel
- **newdata**: predictor or exogenous variables
- **type**: see predict.glm. Default is "response". See predict.glm.
- **previous**: logical vector containing previously occurred states.
- **endogenous**: String vector containing the name of the endogenous variables. It is used if the endogenous variables are more than one, otherwise is set NULL (Default).
- **...**: further arguments
predict.PrecipitationOccurrenceModel

origin_newdata character string containing the date corresponding the first row of newdata
precipitation.value.random.generation logical value. If it is FALSE (Default) the method predict.PrecipitationAmountModel returns conditioned random values, otherwise these values are converted to precipitation values through their observed non-parametric distributions.

Value
A vector or a data frame reporting predicted time series for each station.

See Also
predict.glm,PrecipitationOccurrenceModel
predict.glm,predict.glm,PrecipitationOccurrenceModel,PrecipitationAmountModel

Examples

library(RGENERATEPREC)
data(trentino)

year_min <- 1961
year_max <- 1990

period <- PRECIPITATION$year>=year_min & PRECIPITATION$year<=year_max
period_temp <- TEMPERATURE_MAX$year>=year_min & TEMPERATURE_MAX$year<=year_max

prec_mes <- PRECIPITATION[period,]
Tx_mes <- TEMPERATURE_MAX[period_temp,]
Tn_mes <- TEMPERATURE_MIN[period_temp,]
accepted <- array(TRUE,length(names(prec_mes)))
names(accepted) <- names(prec_mes)
for (it in names(prec_mes)) {
  acc <- TRUE
  acc <- (length(which(!is.na(Tx_mes[,it])))==length(Tx_mes[,it]))
  acc <- (length(which(!is.na(Tn_mes[,it])))==length(Tn_mes[,it])) & acc
  accepted[it] <- (length(which(!is.na(prec_mes[,it])))==length(prec_mes[,it])) & acc
}

valmin <- 1.0
prec_mes <- prec_mes[,accepted]

Tx_mes <- Tx_mes[,accepted]
Tn_mes <- Tn_mes[,accepted]
origin <- paste(year_min,1,1,sep=-")

prec_occurrence_mes <- prec_mes>=valmin
station <- names(prec_mes)[!(names(prec_mes) %in% c("day","month","year"))]
it <- station[2]
vect <- Tx_mes[,it]-Tn_mes[,it]
months <- factor(prec_mes$month)
model <- PrecipitationOccurrenceModel(x=prec_mes[,it],exogen=vect,monthly.factor=months)
probs <- predict(model)

nday <- 3.0
vect_new <- array(1.0,nday)
months_new <- array(1,nday)
row_test <- 2000:2007
newdata <- model$predictor[row_test,]
probs2 <- predict(model,newdata=newdata)
probs[row_test]==probs2
###

prec_occurrence_mes <- prec_mes>=valmin
station <- names(prec_mes)[!(names(prec_mes) %in% c("day","month","year"))]

station <- station[1:4] ## reduced the dataset!!!
Tx_mes <- Tx_mes[,station]
Tn_mes <- Tn_mes[,station]
prec_mes <- prec_mes[,station]
exogen <- Tx_mes-Tn_mes
months <- factor(prec_mes$month)

### Not Run
### Please uncomment the following lines to run them

model_multisite <- PrecipitationOccurrenceMultiSiteModel(x=prec_mes,
exogen=exogen,origin=origin,multisite_type="wilks")

model_multisite_logit <- PrecipitationOccurrenceMultiSiteModel(x=prec_mes,
exogen=exogen,origin=origin,multisite_type="logit")

probs_multimodel <- predict(model_multisite_logit)
Index

blockmatrix, 5

CCGamma, 2, 3, 5, 22, 23
CCGammaToBlockmatrix, 4, 5
continuity_ratio, 2–5
dw.spell, 6
generate, 8–10, 17
generate
  (generate.PrecipitationOccurrenceModel), 8
generate.PrecipitationOccurrenceModel, 8
glm, 20

nearPD, 15
normalCopula, 14–16
normalizeGaussian_severalstations, 17
nwetdays, 11

omega, 2, 4, 5, 13, 14–16
omega_inv, 2–5, 14, 16, 22
omega_root, 15, 16

pcopula, 14–16
PrecipitationAmountModel, 17, 25
PrecipitationOccurrenceModel, 9, 10, 19,
  22–25
PrecipitationOccurrenceMultiSiteModel,
  9, 10, 21
predict
  (predict.PrecipitationOccurrenceModel), 24
predict.glm, 10, 24, 25
predict.PrecipitationAmountModel, 17
predict.PrecipitationOccurrenceModel,
  9, 24

uniroot, 15