Package 'depPPR'

April 23, 2020

Type Package

Title depPPR - Dependence Post-processing in R

Version 0.1.0

Author Kate Saunders

Maintainer Kate Saunders <K.R.Saunders@tudelft.nl>

Description Package for restoring dependence after to univariate forecasts that have been statistically post-processed.

License General licence

Encoding UTF-8

LazyData true

Depends tidyverse

RoxygenNote 6.1.1

R topics documented:

`apply_ecc_template` *Performs empirical copula coupling*

Description

Reshuffles univariate forecasts so that the multivariate forecast inherits the dependence structure the raw ensemble.

Usage

```
apply_ecc_template(X_raw, Y_forecast)
```

Arguments

X_raw	is a matrix where the columns correspond to the raw ensemble members
Y_forecast	is a matrix where the entries correspond to the univariate forecasts

Details

Univariate forecasts are generated using `sample_ecc_members_norm()`. These univariate forecasts are then reshuffled using a dependence template inherited from the raw ensemble, `X_raw`. This process is known as Empircal Copula Coupling (ECC). The type of ECC depends on the sampling type used to generate the members in `sample_ecc_members_norm`.

This function is functionally the same as `schaake_shuffle()`, but to be consistent with the formulation in the orginal papers has different inputs.

Value

a matrix where the columns corresponds to the post-processed multivariate forecasts

Author(s)

Kate Saunders and Kirien Whan

References

Schefzik, Roman, Thordis L. Thorarinsdottir, and Tilmann Gneiting. "Uncertainty quantification in complex simulation models using ensemble copula coupling." Statistical science 28.4 (2013): 616-640.

Hu, Yiming, et al. "A stratified sampling approach for improved sampling from a calibrated ensemble forecast distribution." Journal of Hydrometeorology 17.9 (2016): 2405-2417.

Examples

```
# code for this example was based on the function vs_sample()
# in the scoringRules package

d <- 3  # number of dimensions
m <- 5  # number of ensemble members
mu0 <- rep(0, d)
mu <- rep(1, d)
S0 <- S <- diag(d)
S[S==0] <- 0.1
S0[S0==0] <- 0.2

# generate samples from multivariate normal distributions
obs <- drop(mu0 + rnorm(d) %*% chol(S0))
raw_ensemble <- replicate(m, drop(mu + rnorm(d) %*% chol(S)))

pars = data.frame(mu = mu0, sigma = rep(1, d))
draw_type = 'R'
univariate_forecast <- sample_ecc_members_norm(num_members = m,  pars,  draw_type)
```

```
apply_ecc_template(raw_ensemble, univariate_forecast)
```

`create_member_names` *Creates member names*

Description

Takes a prefix string, ie "fc" and creates member reference names, ie, "fc001" "fc002"

Usage

```
create_member_names(prefix_string, num_members, width)
```

Arguments

`num_members`	integer giving the number of ensemble members
`width`	(optional) this is the width for zero padding with the prefix string (default is 3)
`M`	prefix string

Value

a vector containing the names of the ensemble members

Author(s)

Kate Saunders and Kirien Whan

Examples

```
create_member_names(prefix_string = "fc", num_members = 10)
create_member_names(prefix_string = "fc", num_members = 10, width = 2)
```

`get_ecc_quantiles` *Creates a vector of quantiles*

Description

This function produces qunatiles for ECC-R (random qunatiles), ECC-Q (uniform quantiles) or ECC-S (jittered quantiles).

Usage

```
get_ecc_quantiles(m, ecc_type)
```

Arguments

`m`	number of members in the ensemble (must be an integer)
`ecc_type`	one of the characters `'R'`, `'Q'`, `'Q1'`, or `'S'`. This character corresponds to the desired ECC sampling method. There are two types of equi-spaced quantiles, 'Q' and 'Q1'.

Details

If the `ecc_type` is `'R'` then quantiles are randomly sampled. If the `ecc_type` is `'Q'` then qunatiles are equally sampled. #' If the `ecc_type` is `'Q1'` then quantiles are equally sampled (Bröck et al. 2012) If the `ecc_type` is `'S'` then the quantiles are jittered.

Value

a vector of sample quantiles

Author(s)

Kate Saunders and Kirien Whan

References

Schefzik, Roman, Thordis L. Thorarinsdottir, and Tilmann Gneiting. "Uncertainty quantification in complex simulation models using ensemble copula coupling." Statistical science 28.4 (2013): 616-640.

Hu, Yiming, et al. "A stratified sampling approach for improved sampling from a calibrated ensemble forecast distribution." Journal of Hydrometeorology 17.9 (2016): 2405-2417.

See Also

`sample_ecc_members`

Examples

```
get_ecc_quantiles(3, "R")
get_ecc_quantiles(4, "Q")
get_ecc_quantiles(5, "S")
```

`hello` *Hello, World!*

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

`interpolate_missing_values` *Interpolates missing observations*

Description

Occasionally missing observations are present in observations sampled that are to be used in the Schaake shuffle. If only a few of observations are missing if can be useful to interpolate the missing values, particularly if only a few dates are available for sampling.

Usage

```
interpolate_missing_values(M)
```

Arguments

`M`	is a matrix where columns corresponds to different sample observations

Details

For each column with a missing observation, the next closest column is found in terms of the smallest mean square error. If this close column has a valid observation in the same row as the missing observation, then this value is used to interpolate the missing observation. This is a simple method of data imputation, M can be subsetted in such a way to improve the reliability of the imputed value. Care should be taken that the reason the observation was missing was not for a systematic reason.

Value

M with any missing values interpolated

Author(s)

Kate Saunders and Kirien Whan

Examples

```
M = matrix(c(1,2,3,1,NA,3,4,5,6), nrow = 3)
interpolate_missing_values(M)

M = matrix(c(-1,-2,-3,1,NA,3,4,5,6), nrow = 3)
interpolate_missing_values(M)

M_all = matrix(c(1, 2, 3, 7, 10, 1, NA, 3, 4, 5, 2, 3, 4, 5, 6), nrow = 5)
interpolate_missing_values(M_all)
interpolate_missing_values(M_all[1:3, ])
```

reorder_members — *Reorders ensemble members*

Description

Reorders the row entries of a matrix using a template based on order statistics

Usage

```
reorder_members(X, B)
```

Arguments

X	is a matrix where the columns correspond to multivariate forecasts.
B	is a matrix with common dimension to X, and contains order statsitics for reshuffling

Details

Each columns of X corresponds to an ensemble member. The order statistics in B are generated from climatologically similar days to the forecast day. This function is internal and used within `schaake_shuffle()`

Value

a reshuffled version of matrix according the order statistics given in B.

Author(s)

Kate Saunders and Kirien Whan

Examples

```
X = matrix(c(2,1,3, 5,6,7), nrow = 2, byrow = TRUE)
B = matrix(c(2,1,3, 3,2,1), nrow = 2, byrow = TRUE)
reorder_members(X, B)
```

sample_ecc_members — *Samples ensemble members from a normal distribution*

Description

This functions samples ensembe members from the provided distribution function. The type of sampling can be one of ECC-R (random), ECC-Q (uniform quantiles) or ECC-S (jittered quantiles).

Usage

```
sample_ecc_members(num_members, function_type, pars, ecc_type)
```

Arguments

num_members	number of members in the ensemble (must be an integer)
function_type	a function to simulate the members from
pars	a data frame with named columns corresponding to parameters
ecc_type	one of the characters, 'R', 'Q' or 'S'. This character corresponds to the desired ECC sampling method

Details

If the ecc_type is 'R' then the function_type should be for random sampling of quantiles, ie. `rnorm`. If the ecc_type is 'Q' or 'S' then the function_type should be for quantile sampling, ie `qnorm`.

This function uses get_ecc_quantiles for quantile sampling.

Value

a matrix where the columns give the sampled ensemble members

Author(s)

Kate Saunders and Kirien Whan

References

Schefzik, Roman, Thordis L. Thorarinsdottir, and Tilmann Gneiting. "Uncertainty quantification in complex simulation models using ensemble copula coupling." Statistical science 28.4 (2013): 616-640.

Hu, Yiming, et al. "A stratified sampling approach for improved sampling from a calibrated ensemble forecast distribution." Journal of Hydrometeorology 17.9 (2016): 2405-2417.

See Also

get_ecc_quantiles and reorder_members

Examples

```
num_members = 5
pars <- data.frame(mean = 0, sd = 1)
sample_ecc_members(num_members,  rnorm, pars, 'R')
sample_ecc_members(num_members, qnorm,  pars,  'Q')
sample_ecc_members(num_members, qnorm,  pars, 'S')

pars <- data.frame(mean = c(0,10), sd = c(1,1))
sample_ecc_members(num_members,  rnorm, pars, 'R')

num_members = 4
pars <- data.frame(rate = c(1,2,3))
sample_ecc_members(num_members, rexp,  pars, 'R')
```

`sample_schaake_dates` *Samples dates for schaake shuffle*

Description

For a given date, this function samples dates within a surrounding window across different years. These dates can be used in the Schaake shuffle to generate a dependence template that is based on climatology.

Usage

```
sample_schaake_dates(num_draws, dates, date_val, window = 7)
```

Arguments

`num_draws`	number of dates to samples
`dates`	vector of all possible dates that will be cross-referenced with the climate window
`date_val`	date of the observation for which a similar climatology is required
`window`	integer that gives the radius of the date window, `date_val` +- `window` (unit is days)

Details

This function assumes the window of interest is in given in days

Value

a vector of length `num_draws` that gives the sampled dates

Author(s)

Kate Saunders and Kirien Whan

References

Clark, Martyn, et al. "The Schaake shuffle: A method for reconstructing space–time variability in forecasted precipitation and temperature fields." Journal of Hydrometeorology 5.1 (2004): 243-262.

Examples

```
date_val = lubridate::as_date("2019-01-01")
dates = seq(lubridate::as_date("2018-01-01"), lubridate::as_date("2020-01-01"), by = "days")

set.seed(1)
sampled_dates <- sample_schaake_dates(num_draws = 3, dates = dates, date_val = date_val,  window = 7)
sampled_dates

set.seed(1)
new_dates = setdiff(dates, sampled_dates) %>% lubridate::as_date()
new_sampled_dates <- sample_schaake_dates(num_draws = 3, dates = new_dates, date_val = date_val, window = 7)
new_sampled_dates
```

schaake_shuffle *Performs the schaake shuffle*

Description

For a forecast that has been subject to univariate post-processing the members need to be reshuffled to so that the forecast has the correct dependence structure. This function does the reshuffling using a dependence structure that is inherited from days that have are climatologically similar.

Usage

```
schaake_shuffle(X, Y)
```

Arguments

X	is a matrix where the columns correspond to multivariate forecasts.
Y	is a matrix where the columns give climatologically similar observations to that of the forecast day.

Details

For the input matrix X, the number of rows correspond to the dimension of the multivariate forecast and the number of columns corresponds the ensemble members. The dimension of Y must correspond to X. To get climatologically similar days use the function `sample_schaake_dates()`.

No missing values should be present in Y. Data imputation or date resampling should be used.

This function is functionally the same as `apply_ecc_template()`, but to be consistent with the formulation in the orginal papers has different inputs.

Value

a matrix where the forecasts in X have been reshuffled according to the dependence inherited from the climatological template in Y.

Author(s)

Kate Saunders and Kirien Whan

References

Clark, Martyn, et al. "The Schaake shuffle: A method for reconstructing space–time variability in forecasted precipitation and temperature fields." Journal of Hydrometeorology 5.1 (2004): 243-262.

Examples

```
# code for this example was based on the function vs_sample()
# in the scoringRules package

d <- 3  # number of dimensions
m <- 5  # number of samples from multivariate forecast distribution

mu0 <- rep(0, d)
mu <- rep(1, d)
S0 <- S <- diag(d)
S[S==0] <- 0.1
S0[S0==0] <- 0.2

# generate samples from multivariate normal distributions
obs <- drop(mu0 + rnorm(d) %*% chol(S0))
climate_example <- replicate(m, drop(mu + rnorm(d) %*% chol(S)))

forecast_example <- matrix(mu0 + rnorm(d*m), nrow = d, ncol = m)

schaake_shuffle(X = forecast_example, Y = climate_example)
```

---

shuffle_members *Function to rank, order or sort ensemble members*

Description

Applies one of the functions rank, order or sort to the ensemble members.

Usage

```
shuffle_members(M, type, ...)
```

Arguments

M	is a matrix where the columns correspond to multivariate forecasts.
type	is a string of either 'rank', 'order' or 'sort'. This operation is applied to the rows

Details

No missing values should be present in M.

This function is used within `schaake_shuffle()` and `ecc()`.

Value

a matrix where the ensemble members in M have been altered according to the function

Author(s)

Kate Saunders and Kirien Whan

Examples

```
M = matrix(c(2,3,1, 5,6,7), nrow = 2, byrow = TRUE)
shuffle_members(M, 'rank')
shuffle_members(M, 'order')
shuffle_members(M, 'sort')
```

Index