Package 'copulaSim'

August 6, 2022

Type Package

Title Virtual Patient Simulation by Copula Invariance Property

Version 0.0.1

Date 2022-08-06

Maintainer Pei-Shan Yen <peishan0824@gmail.com>

Description To optimize clinical trial designs and data analysis methods consistently through trial simulation, we need to simulate multivariate mixed-type virtual patient data independent of designs and analysis methods under evaluation. To make the outcome of optimization more realistic, relevant empirical patient level data should be utilized when it's available. However, a few problems arise in simulating trials based on small empirical data, where the underlying marginal distributions and their dependence structure cannot be understood or verified thoroughly due to the limited sample size. To resolve this issue, we use the copula invariance property, which can generate the joint distribution without making a strong parametric assumption. The function copula.sim can generate virtual patient data with optional data validation methods that are based on energy distance and ball divergence measurement. The function compare.copula.sim can conduct comparison of marginal mean and covariance of simulated data. To simulate patient-level data from a hypothetical treatment arm that would perform differently from the observed data, the function new.arm.copula.sim can be used to generate new multivariate data with the same dependence structure of the original data but with a shifted mean vector.

License MIT + file LICENSE

Depends R (>= 4.0.0)

Imports dplyr (>= 1.0.0),
magrittr (>= 1.5),
mvtnorm (>= 1.0-12),
rlang,
stats,
tibble,
utils

Suggests rmarkdown,
knitr,
ggplot2,
testthat (>= 3.1.1),
Ball (>= 1.3.0),
energy (>= 1.7-0),

URL https://github.com/psyen0824/copulaSim

VignetteBuilder knitr

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

R topics documented:

copulaSim-package   *copulaSim: Virtual Patient Simulation by Copula Invariance Property*

Description

To optimize clinical trial designs and data analysis methods consistently through trial simulation, we need to simulate multivariate mixed-type virtual patient data independent of designs and analysis methods under evaluation. To make the outcome of optimization more realistic, relevant empirical patient level data should be utilized when it's available. However, a few problems arise in simulating trials based on small empirical data, where the underlying marginal distributions and their dependence structure cannot be understood or verified thoroughly due to the limited sample size. To resolve this issue, we use the copula invariance property, which can generate the joint distribution without making a strong parametric assumption. The function copula.sim can generate virtual patient data with optional data validation methods that are based on energy distance and ball divergence measurement. The function compare.copula.sim can conduct comparison of marginal mean and covariance of simulated data. To simulate patient-level data from a hypothetical treatment arm that would perform differently from the observed data, the function new.arm.copula.sim can be used to generate new multivariate data with the same dependence structure of the original data but with a shifted mean vector.

Author(s)

Maintainer: Pei-Shan Yen <peishan0824@gmail.com> (ORCID)

Other contributors:

- Xuemin Gu <xuemin.gu@abbvie.com> [contributor]
- Jenny Jiao <jenny.jiao@abbvie.com> [contributor]
- Jane Zhang <jane.zhang@abbvie.com> [contributor]

See Also

Useful links:

- https://github.com/psyen0824/copulaSim

compare.copula.sim	*Performing the comparison between empirical data and multiple simulated datasets.*

Description

Performing the comparison between empirical data and multiple simulated datasets.

Usage

```
compare.copula.sim(object)
```

Arguments

object   A copula.sim object for the comparison.

Value

Returned the comparison of marginal parameter and covariance.

1. mean.comparison: comparison between empirical marginal mean and average value of simulated marginal mean. (1) simu.mean: average value of simulated mean (2) simu.sd: average value of simulated standard error (3) simu.mean.low.lim: lower limit of 95% percentile confidence interval (4) simu.mean.upp.lim: upper limit of 95% percentile confidence interval (5) simu.mean.RB: relative bias (6) simu.mean.SB: standardized bias (7) simu.mean.RMSE: root mean square error

2. cov.comparison: comparison between empirical covariance and average value of simulated covariance

Author(s)

Pei-Shan Yen, Xuemin Gu

copula.sim	*To generate simulated datasets from empirical data by utilizing the copula invariance property.*

Description

Based on the empirical data, generating simulated datasets through the copula invariance property.

Usage

```
copula.sim(
  data.input,
  id.vec,
  arm.vec,
  n.patient,
  n.simulation,
  seed = NULL,
```

```
    validation.type = "none",
    validation.sig.lvl = 0.05,
    rmvnorm.matrix.decomp.method = "svd",
    verbose = TRUE
)
```

Arguments

`data.input`	The empirical patient-level data to be used to simulate new virtual patient data.
`id.vec`	The ID for individual patient in the input data.
`arm.vec`	The column to identify the arm in clinical trial.
`n.patient`	The targeted number of patients in each simulated dataset.
`n.simulation`	The number of simulated datasets.
`seed`	The random seed. Default is NULL to use the current seed.
`validation.type`	
	A string to specify the hypothesis test used to detect the difference between input data and the simulated data. Default is "none". Possible methods are energy distance ("energy") and ball divergence ("ball"). The R packages "energy" and "Ball" are needed.
`validation.sig.lvl`	
	The significant level (alpha) value for the hypothesis test.
`rmvnorm.matrix.decomp.method`	
	The method to do the matrix decomposition used in the function `rmvnorm`. Default is "svd".
`verbose`	A logical value to specify whether to print message for simulation process or not.

Value

A copula.sim object with four elements.

1. data.input: empirical data (wide-form)

2. data.input.long: empirical data (long-form)

3. data.transform: quantile transformation of data.input

4. data.simul: simulated data

Author(s)

Pei-Shan Yen, Xuemin Gu

References

Sklar, A. (1959). Functions de repartition an dimensionset leursmarges., Paris: PublInst Stat.

Nelsen, R. B. (2007). An introduction to copulas. Springer Science & Business Media.

Ross, S. M. (2013). Simulation. Academic Press.

Examples

```
library(copulaSim)

## Generate Empirical Data
 # Assume the 2-arm, 5-dimensional empirical data follows multivariate normal data.
library(mvtnorm)
arm1 <- rmvnorm(n = 40, mean  = rep(10, 5), sigma = diag(5) + 0.5)
arm2 <- rmvnorm(n = 40, mean  = rep(12, 5), sigma = diag(5) + 0.5)
test_data <- as.data.frame(cbind(1:80, rep(1:2, each = 40), rbind(arm1, arm2)))
colnames(test_data) <- c("id","arm",paste0("time_", 1:5))

## Generate 100 simulated datasets
copula.sim(data.input = test_data[,-c(1,2)], id.vec = test_data$id, arm.vec = test_data$arm,
n.patient = 100 , n.simulation = 100, seed = 2022)
```

---

data.diff.test	*Performing the hypothesis test to compare the difference between the empirical data and the simulated data*

Description

Performing the hypothesis test to compare the difference between the empirical data and the simulated data

Usage

```
data.diff.test(x, y, test.method)
```

Arguments

x	A numeric matrix.
y	A numeric matrix which is compared to x.
test.method	A string to specify the hypothesis test used to detect the difference between input data and the simulated data. Default is "none". Possible methods are energy distance ("energy") and ball divergence ("ball"). The R packages "energy" and "Ball" are needed.

Value

A list with two elements.

1. p.value: the p-value of the hypothesis test.

2. test.result: the returned object of the hypothesis test.

---

ecdf.inv   *Obtaining the inverse of marginal empirical cumulative distribution
                                 (ECDF)*

Description

Obtaining the inverse of marginal empirical cumulative distribution (ECDF)

Usage

```
ecdf.inv(x, p, sort.flag = TRUE)
```

Arguments

x   A vector of numbers which is the marginal empirical data.

p   A vector of numbers which is the probability of the simulated data.

sort.flag   A logical value to specify whether to sort the output data.

Value

The inverse values of p based on ECDF of x.

Examples

```
ecdf.inv(0:10, c(0.25, 0.75))
ecdf.inv(0:10, c(0.25, 0.75), FALSE)
```

---

extract.data.sim   *Converting data.simul in a copula.sim object into a list of wide-form
                                 matrices*

Description

Converting data.simul in a copula.sim object into a list of wide-form matrices

Usage

```
extract.data.sim(object)
```

Arguments

object   A copula object.

Value

A list of matrices for simulated data.

new.arm.copula.sim	*Simulating new multivariate datasets with shifted mean vector from existing empirical data*

Description

Simulating new multivariate datasets with shifted mean vector from existing empirical data

Usage

```
new.arm.copula.sim(
  data.input,
  id.vec,
  arm.vec,
  shift.vec.list,
  n.patient,
  n.simulation,
  seed = NULL,
  validation.type = "none",
  validation.sig.lvl = 0.05,
  rmvnorm.matrix.decomp.method = "svd",
  verbose = TRUE
)
```

Arguments

data.input, id.vec, arm.vec, n.patient, n.simulation, seed
                 Please refer to the function copula.sim.

shift.vec.list A list of numeric vectors to specify the mean-shifted values for new arms.

validation.type, validation.sig.lvl, rmvnorm.matrix.decomp.method, verbose
                 Please refer to the function copula.sim.

Value

Please refer to the function copula.sim.

Author(s)

Pei-Shan Yen, Xuemin Gu, Jenny Jiao, Jane Zhang

Examples

```
library(copulaSim)

## Generate Empirical Data
 # Assume that the single-arm, 3-dimensional empirical data follows multivariate normal data
library(mvtnorm)
arm1 <- rmvnorm(n = 80, mean = c(10,10.5,11), sigma = diag(3) + 0.5)
test_data <- as.data.frame(cbind(1:80, rep(1,80), arm1))
colnames(test_data) <- c("id", "arm", paste0("time_", 1:3))

## Generate 1 simulated datasets with one empirical arm and two new-arm.
```

```
## The mean difference between empirical arm and
 # (i) the 1st new arm is assumed to be 2.5, 2.55, and 2.6 at each time point
 # (ii) the 2nd new arm is assumed to be 4.5, 4.55, and 4.6 at each time point
new.arm.copula.sim(data.input = test_data[,-c(1,2)],
  id.vec = test_data$id, arm.vec = test_data$arm,
  n.patient = 100 , n.simulation = 1, seed = 2022,
  shift.vec.list = list(c(2.5,2.55,2.6), c(4.5,4.55,4.6)))
```

Index