Package 'copent'

July 22, 2025

Version 0.5

Date 2024-06-08

Title Estimating Copula Entropy and Transfer Entropy

Author MA Jian [aut, cre]

Maintainer MA Jian <majian03@gmail.com>

Depends R (>= 2.7.0)

Imports stats, parallel

Suggests mnormt

Description The nonparametric methods for estimating copula entropy, transfer entropy, and the statistics for multivariate normality test and two-sample test are implemented. The methods for estimating transfer entropy and the statistics for multivariate normality test and two-sample test are based on the method for estimating copula entropy. The method for change point detection with copula entropy based two-sample test is also implemented. Please refer to Ma and Sun (2011) <doi:10.1016/S1007-0214(11)70008-6>, Ma (2019) <doi:10.48550/arXiv.1910.04375>, Ma (2022) <doi:10.48550/arXiv.2206.05956>, Ma (2023) <doi:10.4855
formation.

License GPL (>= 2)

URL https://github.com/majianthu/copent

NeedsCompilation no

Repository CRAN

Date/Publication 2024-06-07 23:40:02 UTC

Contents

`ci` *Conditional independence test with copula entropy*

Description

Testing conditional independence between (x,y) conditional on z with copula entropy.

Usage

```
ci(x,y,z,k=3,dt=2)
```

Arguments

`x`	the data with 1 row
`y`	the data with 1 row
`z`	the data with 1 row
`k`	kth nearest neighbour, default = 3
`dt`	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance

Details

This program involves testing conditional independence between (**x,y**) conditional on **z** with copula entropy nonparametrically. It was proposed in Ma (2019).

The algorithm composes of two simple steps: estimating three copula entropy terms with `copent` and then calculate the test statistic.

The argument **x,y,z** are for the data with 1 row and same length as samples from random variables. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance).

Value

The function returns the value of the test statistic of conditional independence.

References

Ma, Jian. Estimating Transfer Entropy via Copula Entropy. arXiv preprint arXiv:1910.04375, 2019.

Examples

```
library(copent)
library(mnormt)
rho1 <- 0.5
rho2 <- 0.6
rho3 <- 0.5
sigma <- matrix(c(1,rho1,rho2,rho1,1,rho3,rho2,rho3,1),3,3)
x <- rmnorm(500,c(0,0,0),sigma)
ci1 <- ci(x[,1],x[,2],x[,3])
```

`construct_empirical_copula` *Construct empirical copula by rank statistic*

Description

Construct empirical copula by rank statistic.

Usage

```
construct_empirical_copula(x)
```

Arguments

x	the data with each row as a sample

Details

This program involves estimating empirical copula from data by rank statistic nonparametrically. It was proposed in Ma and Sun (2008, 2011). The algorithm is the first step of estimating copula entropy `copent`.

The argument **x** is for the data with each row as a sample from random variables.

Value

The function returns the estimated empirical copula of data **x**.

References

Ma, J., & Sun, Z. (2011). Mutual information is copula entropy. *Tsinghua Science & Technology*, **16**(1): 51-54. See also *ArXiv preprint*, arXiv: 0808.0845, 2008.

Examples

```
library(mnormt)
rho <- 0.5
sigma <- matrix(c(1,rho,rho,1),2,2)
x <- rmnorm(500,c(0,0),sigma)
xc1 <- construct_empirical_copula(x)
```

`copent` *Estimating copula entropy*

Description

Estimating copula entropy nonparametrically.

Usage

```
copent(x,k=3,dt=2)
```

Arguments

`x`	data with each row as a sample
`k`	kth nearest neighbour, default = 3
`dt`	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance

Details

This program involves estimating copula entropy from data nonparametrically. It was proposed in Ma and Sun (2008, 2011).

The algorithm composes of two simple steps: estimating empirical copula by rank statistic using `construct_empirical_copula` and then estimating copula entropy with kNN method using `entknn` proposed in Kraskov et al (2004).

The argument **x** is for the data with each row as a sample from random variables. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance).

Copula Entropy is proved to be equivalent to negative mutual information so this program can also be used to estimate multivariate mutual information.

Value

The function returns *negative* value of copula entropy of data **x**.

References

Ma, J., & Sun, Z. (2011). Mutual information is copula entropy. *Tsinghua Science & Technology*, **16**(1): 51-54. See also arXiv preprint arXiv:0808.0845, 2008.

Kraskov, A., St\"ogbauer, H., & Grassberger, P. (2004). Estimating Mutual Information. *Physical Review E*, **69**(6), 66138.

Examples

```
library(mnormt)
rho <- 0.5
sigma <- matrix(c(1,rho,rho,1),2,2)
x <- rmnorm(500,c(0,0),sigma)
ce1 <- copent(x,3,2)
```

---

cpd *Single change point detection with copula entropy based two-sample test*

Description

Single change point detection with copula entropy based two-sample test.

Usage

```
cpd(x,thd=0.13,n=15,k=3,dt=2,ncores=0)
```

Arguments

x	data with each row as a sample of d-dimensional random variables
thd	threshold of the statistic of two-sample test for detecting a change point, default = 0.13
n	the argument used by two-sample test, default = 15
k	kth nearest neighbour, default = 3
dt	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance. default = 2
ncores	number of cores to be used for parallel computing, default = 0 for all the cores

Details

This program involves detecting single change point in univariate or multivariate time series data with copula entropy based two-sample test. It was proposed in Ma (2024), in which a group of two-sample tests are performed on time series data and the change point is considered to be associated with the maximum of the statistics of all the tests.

The argument **x** is for the data with each row as a sample of d-dimensional random variables. The argument **thd** is for the threshold of the statistic of two-sample test for detecting a change point. If

the maximum of the statistics of all the two-sample tests is below the threshold, no change point is detected. The argument **n** is the argument used by the two-sample test function `tst`. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance). The argument **ncores** is for the number of cores to be used for parallel computing. If the default 0 is used, then all the cores will be used.

Value

The function returns a list containing

`stats`	the estimated statistics of all the two-sample tests
`maxstat`	the maximum of the estimated statistics
`pos`	the change point detected

References

Ma, Jian. Change Point Detection with Copula Entropy based Two-Sample Test. arXiv preprint arXiv:2403.07892, 2024.

Examples

```
x = c(rnorm(15,0,1),rnorm(15,0,10))
cpd(x,thd=0.15,ncores=2)
```

`entknn` *Estimating entropy from data with kNN method*

Description

Estimating entropy from data with kNN method.

Usage

```
entknn(x,k=3,dt=2)
```

Arguments

`x`	the data with each row as a sample
`k`	kth nearest neighbour, default = 3
`dt`	the type of distance between samples, = 1 for Eclidean distance; other for Maximum distance

Details

This program involves estimating entropy from data by kNN method. It was proposed in Kraskov et al (2004). The algorithm is the second step of estimating copula entropy `copent`.

The argument **x** is for the data with each row as a sample from random variables. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance).

Value

The function returns the estimated entropy value of data **x**.

References

Kraskov, A., St\"ogbauer, H., & Grassberger, P. (2004). Estimating Mutual Information. *Physical Review E*, **69**(6), 66138.

Examples

```
library(mnormt)
rho <- 0.5
sigma <- matrix(c(1,rho,rho,1),2,2)
x <- rmnorm(500,c(0,0),sigma)
xent1 <- entknn(x)
```

`mcpd` *Multiple change point detection with copula entropy based two-sample test*

Description

Multiple change point detection with copula entropy based two-sample test.

Usage

```
mcpd(x,maxp=5,thd=0.13,minseglen=10,n=15,k=3,dt=2,ncores=0)
```

Arguments

`x`	data with each row as a sample of d-dimensional random variables
`maxp`	maximal number of change points, default = 5
`thd`	threshold of the statistic of two-sample test for detecting change points, default = 0.13
`minseglen`	minimal length of binary segmentation, default = 10
`n`	the parameter used by two-sample test, default = 15

k	kth nearest neighbour, default = 3
dt	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance
ncores	number of cores to be used for parallel computing, default = 0 for all the cores

Details

This program involves detecting multiple change points in univariate or multivariate time series data with copula entropy based two-sample test. It was proposed in Ma (2024). The method is a combination of binary segmentation and single change point detection implemented in cpd.

The argument **x** is for the data with each row as a sample of d-dimensional random variables. The argument **maxp** is for the maximal number of change points. The argument **thd** is for the threshold of the statistic of two-sample test for detecting a change point used in cpd. The argument **minseglen** is for the minimal length of each segment in binary segmentation. If the length of a segment is shorter than **minseglen**, then no detection will be performed on the segment. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance). The argument **ncores** is for the number of cores to be used for parallel computing. If the default 0 is used, then all the cores will be used.

Value

The function returns a list containing

maxstat	the maximal statistics of the detected change points
pos	the change points detected

References

Ma, Jian. Change Point Detection with Copula Entropy based Two-Sample Test. arXiv preprint arXiv:2403.07892, 2024.

Examples

```
x = c(rnorm(15,0,1),rnorm(10,0,10),rnorm(10,0,1))
mcpd(x,thd=0.15,ncores=2)
```

---

mvnt *Multivariate normality test with copula entropy*

Description

Estimating the statistic for testing multivariate normality based on copula entropy.

Usage

```
mvnt(x,k=3,dt=2)
```

Arguments

x	data with each row as a sample of d-dimensional random variables
k	kth nearest neighbour, default = 3
dt	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance

Details

This program involves estimating the statistic for testing multivariate normality based on copula entropy. It was proposed in Ma (2022). The test statistic is defined as the difference between the copula entropies of unknown distribution and the Gaussian distribution with same covariance.

The argument **x** is for the data with each row as a sample of d-dimensional random variables. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance).

Value

The function returns the statistic for testing multivariate normality of **x**.

References

Ma, Jian. Multivariate Normality Test with Copula Entropy. arXiv preprint arXiv:2206.05956, 2022.

Examples

```
library(mnormt)
rho <- 0.5
sigma <- matrix(c(1,rho,rho,1),2,2)
x <- rmnorm(1000,c(0,0),sigma)
mvnt(x)
```

---

transent *Estimating transfer entropy via copula entropy*

Description

Estimating transfer entropy via copula entropy nonparametrically.

Usage

```
transent(x,y,lag=1,k=3,dt=2)
```

Arguments

x	data with 1 row
y	data with 1 row
lag	time lag, >0
k	kth nearest neighbour, default = 3
dt	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance

Details

This program involves estimating transfer entropy from **y** to **x** with time lag **lag** via copula entropy nonparametrically. It was proposed in Ma (2019).

The algorithm first prepare the data according to **lag**, and then call `ci` for conditional independence testing.

The argument **x,y** are for the data with 1 row as samples from random variables. The argument **lag** is for time lag. The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance).

Value

The function returns the value of transfer entropy from **y** to **x** with time lag **lag**.

References

Ma, Jian. Estimating Transfer Entropy via Copula Entropy. arXiv preprint arXiv:1910.04375, 2019.

Examples

```
library(copent)
num = 300
x = rnorm(num)
y = rnorm(num)
transent(y,x,2)
```

---

tst *Two-sample test with copula entropy*

Description

Estimating the statistic for two-sample test based on copula entropy.

Usage

```
tst(s0,s1,n=12,k=3,dt=2)
```

Arguments

s0, s1	two samples with each row as a sample of d-dimensional random variables
n	repeat time of estimation to reduce estimation bias, default = 12
k	kth nearest neighbour, default = 3
dt	the type of distance between samples, 1 for Eclidean distance; 2 for Maximum distance

Details

This program involves estimating the statistic for non-parametric multivariate two-sample test based on copula entropy. It was proposed in Ma (2023). The test statistic is defined as the difference between the copula entropies of the null hypothesis and the alternative of two-sample test.

The argument **s0,s1** is for the two samples with each row as a sample of d-dimensional random variables. The argument **n** is the repeat time of estimation for reducing the estimation bias (dafault = 12). The argument **k** and **dt** is used in the kNN method for estimating entropy. **k** is for the kth nearest neighbour (default = 3) and **dt** is for the type of distance between samples which has currently two value options (1 for Eclidean distance, and 2(default) for Maximum distance).

Value

The function returns the statistic for two-sample test on **s0,s1**.

References

Ma, Jian. Two-Sample Test with Copula Entropy. arXiv preprint arXiv:2307.07247, 2023.

Examples

```
library(mnormt)
rho <- 0.5
sigma <- matrix(c(1,rho,rho,1),2,2)
s0 <- rmnorm(400,c(0,0),sigma)
s1 <- rmnorm(500,c(5,5),sigma)
tst(s0,s1)
```

Index