

Package: subcopem2D (via r-universe)

November 6, 2024

Type Package

Title Bivariate Empirical Subcopula

Version 1.3

Date 2019-03-18

Author Arturo Erdely

Maintainer Arturo Erdely <arturo.erdely@comunidad.unam.mx>

Description Calculate empirical subcopula and dependence measures from a given bivariate sample, and Bernstein copula approximations.

License GPL-3

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-19 05:40:02 UTC

Contents

Bcopula	1
dependence	3
subcopem	5
subcopemc	7

Index	10
--------------	-----------

Bcopula	<i>Bernstein Copula Approximation</i>
---------	---------------------------------------

Description

Bernstein copula approximation from the empirical subcopula of given bivariate data.

Usage

```
Bcopula(mat.xy, m, both.cont = FALSE, tolimit = 1e-05)
```

Arguments

<code>mat.xy</code>	2-column matrix with bivariate observations of a random vector (X, Y) .
<code>m</code>	integer value of approximation order, where $m = 2, \dots, n$ with n equal to sample size. A recommended value for m would be the minimum between \sqrt{n} and 50.
<code>both.cont</code>	logical value, if TRUE then (X, Y) are considered (both) as continuous random variables, and jittering will be applied to repeated values (if any).
<code>tolimit</code>	tolerance limit in numerical approximation of the inverse of the first partial derivatives of the estimated Bernstein copula.

Details

Each of the random variables X and Y may be of any kind (discrete, continuous, or mixed). NA values are not allowed.

Value

A list containing the following components:

<code>copula</code>	bivariate Bernstein Copula function (BC) of order m
<code>du</code>	bivariate function $\partial BC(u, v)/\partial u$
<code>dv</code>	bivariate function $\partial BC(u, v)/\partial v$
<code>du.inv</code>	inverse of <code>du</code> with respect to v , given u and α (numerical approx)
<code>dv.inv</code>	inverse of <code>dv</code> with respect to u , given v and α (numerical approx)
<code>density</code>	bivariate Bernstein copula density function of order m
<code>bilinearCopula</code>	bivariate function of bilinear approximation of copula
<code>bilinearSubcopula</code>	$(m + 1) \times (m + 1)$ matrix with empirical subcopula values
<code>sample.size</code>	sample size of bivariate observations
<code>order</code>	approximation order m used
<code>both.cont</code>	logical value, TRUE if both variables considered as continuous
<code>tolimit</code>	tolerance limit in numerical approximation of <code>du.inv</code> and <code>dv.inv</code>
<code>subcopemObject</code>	list object with the output from <code>subcopem</code> if <code>both.cont = FALSE</code> or from <code>subcopemc</code> if <code>both.cont = TRUE</code>

Acknowledgement

Development of this code was partially supported by Programa UNAM DGAPA PAPIIT through project IN115817.

Note

If both X and Y are continuous random variables it is faster and better to set `both.cont = TRUE`.

Author(s)

Arturo Erdely <https://sites.google.com/site/arturoerdely>

References

Erdely, A. (2017) *A subcopula based dependence measure*. Kybernetika 53(2), 231-243. DOI: 10.14736/kyb-2017-2-0231

Nelsen, R.B. (2006) *An Introduction to Copulas*. Springer, New York.

Sancetta, A., Satchell, S. (2004) *The Bernstein copula and its applications to modeling and approximations of multivariate distributions*. Econometric Theory 20, 535-562. DOI: 10.1017/S026646660420305X

See Also

[subcopem](#), [subcopemc](#)

Examples

```
## (X,Y) continuous random variables with copula FGM(param = 1)

# Theoretical formulas
FGMcopula <- function(u, v) u*v*(1 + (1 - u)*(1 - v))
dFGM.du <- function(u, v) (2*u - 1)*(v^2) + 2*v*(1 - u)
dFGM.dv <- function(u, v) (2*v - 1)*(u^2) + 2*u*(1 - v)
A1 <- function(u) 2*(1 - u)
A2 <- function(u, z) sqrt(A1(u)^2 - 4*(A1(u) - 1)*z)
dFGM.du.inv <- function(u, z) 2*z/(A1(u) + A2(u, z))
FGMdensity <- function(u, v) 2*(1 - u - v + 2*u*v)

# Simulating FGM observations
n <- 3000
U <- runif(n)
Z <- runif(n)
V <- mapply(dFGM.du.inv, U, Z)

# Applying Bcopula to FGM simulated values
B <- Bcopula(cbind(U, V), 50, TRUE)
str(B)

# Comparing theoretical values versus Bernstein and Bilinear approximations
u <- 0.70; v <- 0.55
FGMcopula(u, v); B[["copula"]](u, v); B[["bilinearCopula"]](u, v)
dFGM.du(u, v); B[["du"]](u, v)
dFGM.dv(u, v); B[["dv"]](u, v)
dFGM.du.inv(u, 0.8); B[["du.inv"]](u, 0.8)
FGMdensity(u, v); B[["density"]](u, v)
```

dependence

Dependence Measures

Description

Calculation of pairwise monotone and supremum dependence, monotone/supremum dependence ratio, and proportion of pairwise NAs.

Usage

```
dependence(mat, cont = NULL, sc.order = 0)
```

Arguments

<code>mat</code>	k -column matrix with n observations of a k -dimensional random vector (NA values are allowed).
<code>cont</code>	vector of column numbers to consider/coerce as continuous random variables (optional).
<code>sc.order</code>	order of subcopula approximation (continuous random variables). If 0 (default) then maximum order $m = n$ is used. Often $m = 50$ is a good recommended value, higher values demand more computing time.

Details

Each of the random variables in the k -dimensional random vector under consideration may be of any kind (discrete, continuous, or mixed). NA values are allowed.

Value

A 3-dimensional array $k \times k \times 4$ with pairwise monotone and supremum dependence, monotone/supremum dependence ratio, and proportion of pairwise NAs.

Note

NA values are allowed.

Author(s)

Arturo Erdely <https://sites.google.com/site/arturoerdely>

References

Erdely, A. (2017) *A subcopula based dependence measure*. Kybernetika 53(2), 231-243. DOI: 10.14736/kyb-2017-2-0231

Nelsen, R.B. (2006) *An Introduction to Copulas*. Springer, New York.

See Also

[subcopem](#), [subcopemc](#)

Examples

```
V <- runif(300) # Continuous Uniform(0,1)
W <- V*(1-V) # Continuous transform of V
# X given V=v as continuous Uniform(0,v)
X <- mapply(runif, rep(1, length(V)), rep(0, length(V)), V)
Y <- 1*(0.2 < X)*(X < 0.6) # Discrete transform of X
Z <- X*(0.1 < X)*(X < 0.9) + 1*(X >= 0.9) # Mixed transform of X
```

```

V[1:10] <- NA # Introducing some NAs
W[3:12] <- NA # Introducing some NAs
Y[5:25] <- NA # Introducing some NAs
vector5D <- cbind(V, W, X, Y, Z) # Matrix of 5-variate observations
# Monotone and supremum dependence, ratio and proportion of NAs:
(depararray <- dependence(vector5D, cont = c(1, 2, 3), 30))
# Pearson's correlations:
cor(vector5D, method = "pearson", use = "pairwise.complete.obs")
# Spearman's correlations:
cor(vector5D, method = "spearman", use = "pairwise.complete.obs")
# Kendall's correlations:
cor(vector5D, method = "kendall", use = "pairwise.complete.obs")
pairs(vector5D) # Matrix of pairwise scatterplots

```

subcopem

Bivariate Empirical Subcopula

Description

Calculation of bivariate empirical subcopula matrix, induced partitions, standardized bivariate sample, and dependence measures for a given bivariate sample.

Usage

```
subcopem(mat.xy, display = FALSE)
```

Arguments

mat.xy	2-column matrix with bivariate observations of a random vector (X, Y) .
display	logical value indicating if graphs and dependence measures should be displayed.

Details

Each of the random variables X and Y may be of any kind (discrete, continuous, or mixed). NA values are not allowed.

Value

A list containing the following components:

depMon	monotone standardized supremum distance in $[-1, 1]$.
depMonNonSTD	monotone non-standardized supremum distance $[min, value, max]$.
depSup	standardized supremum distance in $[0, 1]$.
depSupNonSTD	non-standardized supremum distance $[min, value, max]$.
matrix	matrix with empirical subcopula values.
part1	vector with partition induced by first variable X .
part2	vector with partition induced by second variable Y .

sample.size numeric value of sample size.
 std.sample 2-column matrix with the standardized bivariate sample.
 sample 2-column matrix with the original bivariate sample of (X, Y) .

If `display = TRUE` then the values of `depMon`, `depMonNonSTD`, `depSup`, and `depSupNonSTD` will be displayed, and the following graphs will be generated: marginal histograms of X and Y , scatterplots of the original and the standardized bivariate sample, contour and image bivariate graphs of the empirical subcopula.

Note

If both X and Y are continuous random variables it is faster and better to use [subcopemc](#).

Author(s)

Arturo Erdely <https://sites.google.com/site/arturoerdely>

References

Durante, F. and Sempi, C. (2016) *Principles of Copula Theory*. Taylor and Francis Group, Boca Raton.

Erdely, A. (2017) *A subcopula based dependence measure*. *Kybernetika* 53(2), 231-243. DOI: 10.14736/kyb-2017-2-0231

Nelsen, R.B. (2006) *An Introduction to Copulas*. Springer, New York.

See Also

[subcopemc](#)

Examples

```
## Example 1: Discrete-discrete Poisson positive dependence
n <- 1000 # sample size
X <- rpois(n, 5) # Poisson(parameter = 5)
p <- 2 # another parameter
Y <- mapply(rpois, rep(1, n), 1 + p*X) # creating dependence
XY <- cbind(X, Y) # 2-column matrix with bivariate sample
cor(XY, method = "pearson")[1, 2] # Pearson's correlation
cor(XY, method = "spearman")[1, 2] # Spearman's correlation
cor(XY, method = "kendall")[1, 2] # Kendall's correlation
SC <- subcopem(XY, display = TRUE)
str(SC)
```

```
## Example 2: Continuous-discrete non-monotone dependence
n <- 1000 # sample size
X <- rnorm(n) # Normal(0,1)
Y <- 2*(X > 1) - 1*(X > -1) # Discrete({-1, 0, 1})
XY <- cbind(X, Y) # 2-column matrix with bivariate sample
cor(XY, method = "pearson")[1, 2] # Pearson's correlation
cor(XY, method = "spearman")[1, 2] # Spearman's correlation
```

```
cor(XY, method = "kendall")[1, 2] # Kendall's correlation
SC <- subcopem(XY, display = TRUE)
str(SC)
```

subcopemc

Bivariate Empirical Subcopula of Given Approximation Order

Description

Calculation of bivariate empirical subcopula matrix of given approximation order, induced partitions, standardized bivariate sample, and dependence measures for a given **continuous** bivariate sample.

Usage

```
subcopemc(mat.xy, m = nrow(mat.xy), display = FALSE)
```

Arguments

mat.xy	2-column matrix with bivariate observations of a continuous random vector (X, Y) .
m	integer value of approximation order, where $m = 2, \dots, n$ with n equal to sample size.
display	logical value indicating if graphs and dependence values should be displayed.

Details

Both random variables X and Y must be continuous, and therefore repeated values in the sample are not expected. If found, `jitter` will be applied to break ties. NA values are not allowed.

Value

A list containing the following components:

depMon	monotone standardized supremum distance in $[-1, 1]$.
depMonNonSTD	monotone non-standardized supremum distance $[min, value, max]$.
depSup	standardized supremum distance in $[0, 1]$.
depSupNonSTD	non-standardized supremum distance $[min, value, max]$.
matrix	matrix with empirical subcopula values.
part1	vector with partition induced by first variable X .
part2	vector with partition induced by second variable Y .
sample.size	numeric value of sample size.
order	numeric value of approximation order.
std.sample	2-column matrix with the standardized bivariate sample.

sample 2-column matrix with the original bivariate sample of (X, Y) .

If `display = TRUE` then the values of `depMon`, `depMonNonSTD`, `depSup`, and `depSupNonSTD` will be displayed, and the following graphs will be generated: marginal histograms of X and Y , scatterplots of the original and the standardized bivariate sample, contour and image bivariate graphs of the empirical subcopula.

Note

If approximation order $m > 2000$ calculation may take more than 2 minutes. Usually $m = 50$ would be enough for an acceptable approximation.

Author(s)

Arturo Erdely <https://sites.google.com/site/arturoerdely>

References

Durante, F. and Sempi, C. (2016) *Principles of Copula Theory*. Taylor and Francis Group, Boca Raton.

Erdely, A. (2017) *A subcopula based dependence measure*. *Kybernetika* 53(2), 231-243. DOI: 10.14736/kyb-2017-2-0231

Nelsen, R.B. (2006) *An Introduction to Copulas*. Springer, New York.

See Also

[subcopem](#)

Examples

```
## Example 1: Independent Normal and Gamma

n <- 300 # sample size
X <- rnorm(n) # Normal(0,1)
Y <- rgamma(n, 2, 3) # Gamma(2,3)
XY <- cbind(X, Y) # 2-column matrix with bivariate sample
cor(XY, method = "pearson")[1, 2] # Pearson's correlation
cor(XY, method = "spearman")[1, 2] # Spearman's correlation
cor(XY, method = "kendall")[1, 2] # Kendall's correlation
SC <- subcopemc(XY, , display = TRUE)
str(SC)

## Approximation of order m = 15
SCm15 <- subcopemc(XY, 15, display = TRUE)
str(SCm15)

## Example 2: Non-monotone dependence

n <- 300 # sample size
Theta <- runif(n, 0, 2*pi) # Uniform circular distribution
X <- cos(Theta)
Y <- sin(Theta)
```



```
XY <- cbind(X, Y) # 2-column matrix with bivariate sample
cor(XY, method = "pearson")[1, 2] # Pearson's correlation
cor(XY, method = "spearman")[1, 2] # Spearman's correlation
cor(XY, method = "kendall")[1, 2] # Kendall's correlation
SC <- subcopemc(XY,, display = TRUE)
str(SC)
## Approximation of order m = 15
SCm15 <- subcopemc(XY, 15, display = TRUE)
str(SCm15)
```

Index

* **Bcopula**

Bcopula, [1](#)

* **subcopula**

subcopem, [5](#)

subcopemc, [7](#)

Bcopula, [1](#)

dependence, [3](#)

subcopem, [3](#), [4](#), [5](#), [8](#)

subcopemc, [3](#), [4](#), [6](#), [7](#)