Package ‘vfcp’

October 12, 2022

Type Package
Date 2017-10-24
Title Computation of v Values for U and Copula C(U, v)
Version 1.4.0
Author Josef Brejcha
Maintainer Josef Brejcha <brchjo@gmail.com>
Depends copula, extraDistr, stringr
Suggests knitr, rmarkdown
Description Computation the value of one of two uniformly distributed marginals if the copula probability value is known and the value of the second marginal is also known.
Computation and plotting corresponding cumulative distribution function or survival function.
The numerical definition of a common area limited by lines of the cumulative distribution function and survival function.
Approximate quantification of the probability of this area.
In addition to ‘amh’, the copula dimension may be larger than 2.
License GPL (>= 3)
Encoding UTF-8
LazyData TRUE
RoxygenNote 6.0.1
NeedsCompilation no
Repository CRAN
Date/Publication 2017-10-27 13:02:22 UTC

R topics documented:

vfcp-package ................................................. 2
gentruk ......................................................... 3
kopula ......................................................... 4
prosim ......................................................... 5
Description

Computation of \( v \) when \( u \) and \( C(u, v) \) copula are known. Calculation and plotting of cumulative distribution and survival function when \( u, C(u, v) \) copula and marginal distributions are known. These calculations can be tabulated as option. The numerical definition of a common area limited by lines of the cumulative distribution function and survival function. Approximate quantification of the probability of this area. In addition to ‘amh’, the copula dimension may be larger than 2.

Details

Package: vfcp
Type: Package
Version: 1.4.0
Date: 2017-10-24
License: GPL (>= 3)

Author(s)

Josef Brejcha

Maintainer: Josef Brejcha <brchjo@gmail.com>
References


gentruk

Creating an object for CDF and copula survival

Description

For given inputs, the coordinates of the object defined by the CDF and the survival function for the copula object are created.

Usage

gentruk(tht, fm, C, pro)

Arguments

tht Copula parameter. If fam = "fgm", it must be a vector of size dm*(dm−1)/2 + 1.
fm Family name copula. These can be: "clayton", "gumbel", "frank", "joe", "amh", "fgm".
C Probability value of the copula. Single value.
pro Numeric vector. Its pro[1:k] are upper values of the u. Next pro[-c(1:k)] are then all greater than or equal to 1.

Value

A list with components as trimeze value.

Author(s)

Josef Brejcha
Examples

tht = 0.6

```r
cx = c(0.025, 0.05, 0.1, 0.15, 0.25)
pro = c(0.999999, 0.99999, 0.9999, 0.999, 24, 16, 8, 4)
dm = 2

fam = "fgm"
marg = c("weibull", "betapr")
xo = c(200, 2.75, 16.5, 6.60)
e12 = vfenuo(marg, xo)
p = numeric(length(cx))
x12 = qweibull(0.975, scale = xo[1], shape = xo[2])
y12 = qbetapr(0.975, shape1 = xo[3], shape2 = xo[4])
plot(NULL, NULL, xlim = c(0, x12), ylim = c(0, y12),
    xlab = paste("x, E[x] = ", round(e12[1], 2)),
ylab = paste("y, E[y] = ", round(e12[2], 2)),
    main = mtit)
points(e12[1], e12[2], pch = 20)
abline(h = e12[2], v = e12[1])
grid(col = "grey50")

#===========================
kop2 = kopula(fam, tht, dm)
fmc = c("", "", "clayton", "gumbel", "frank", "joe")
pro = c(0.9999999, 0.999999, 0.999999, 16, 8, 4, 2)
tm3 = list()
tmk = list()
for (k in 1:length(cx)){
    tm3 = gentruk(tht, fm=fam, C=cx[k], pro)
    tmk[[k]] = tm3
}
p = prosim(C = cx, fam, tht, dm, no = 100000)

# =============
xa = c("u")
xa = c("v")
for (k in 1:length(cx)){
    mspx = vfmrg(rdj=marg, i=1, cosi=tmk[[k]]$sp$s1, yo=xo, cdf=TRUE)
    mspx = vfmrg(rdj=marg, i=2, cosi=tmk[[k]]$sp$s2, yo=xo, cdf=TRUE)
    mcpy = vfmrg(rdj=marg, i=2, cosi=tmk[[k]]$cp$c2, yo=xo, cdf=TRUE)
    lines(mspx,mspy, col=k)
    lines(mcpx, mcpy, col = k)
}
legend("topleft", legend = c("C", cx), text.col = c(1, 1:length(cx)),
bty = "n")
legend("topright", legend = c("p", round(p, 4)),
text.col = c(1, 1:length(cx)), bty = "n")
```

kopula

Copula object
Description

Generate the copula object.

Usage

kopula(fam, tht, dm)

Arguments

- **fam**: Family name copula. These can be: "clayton", "gumbel", "frank", "joe", "amh", "fgm".
- **tht**: Copula parameter.
- **dm**: Copula dimension.

Value

Copula object

Author(s)

Josef Brejcha

---

prosim

Monte Carlo method

Description

Probability of the inside of an object as defined by CDF and survival. For this, the Monte Carlo method is used.

Usage

prosim(C, fam, tht, dm, no)

Arguments

- **C**: single numeric; CDF value. Survival value is 1 − CDF.
- **fam**: Family name copula. These can be: "clayton", "gumbel", "frank", "joe", "amh", "fgm".
- **tht**: Copula parameter. If fam = "fgm", it must be a vector of size dm*(dm−1)/2 + 1.
- **dm**: Copula dimension
- **no**: Monte Carlo sample size

Value

Probability
Author(s)

Josef Brejcha

Examples

\begin{verbatim}
tht = 10.6
cx = c(0.05, 0.1, 0.15, 0.25)
pro = c(0.99999, 0.9999, 0.999, 0.99, 24, 16, 8, 4)
dm = 4
fam = "gumbel"
marg = rep(c("weibull", "betaPr"), 2)
xo = rep(c(200, 2.75, 16.5, 6.60), 2)
#=======================================
kop2 = kopula(fam, tht, dm)
fmc = c("", ", "clayton", "gumbel", "frank", "joe")
pro = c(0.999999, 0.99999, 0.9999, 16, 8, 4, 2)
tm3 = list()
tmk = list()
# di = dm*(dm - 1)/2
for (k in 1:length(cx)){
  tm3 = gentruk(tht, fm=fam, C=cx[k], pro)
  tmk[[k]] = tm3
}
np = 5
no = 100000
ncx = length(cx)
p = array(0, c(np*ncx, 2))
colnames(p) = c("C", "p")
k = 0
for (i in 1:length(cx)){
  for (j in 1:np){
    k = k + 1
    p[k, 1] = cx[i]
    p[k, 2] = prosim(C = cx[i], fam, tht, dm, no)
  }
}
plst = list()
print(paste(fam, "dim =", dm, "tht =", tht, "n =", no, "nrep.", np))
for (k in 1:ncx){
  plst[[k]] = summary(p[p[, 1] == cx[k], 2])
  print(paste("cx =", cx[k]))
  print(paste(plst[[k]]))
}
\end{verbatim}

prunikus

The coordinates of the intersection lines of the cumulative distribution function and survival function
Description
The coordinates of the intersection lines of the cumulative distribution function and survival function.

Usage
prunikus(x, y)

Arguments
x Numeric vector of size 4. The horizontal coordinates of opposite points.
y Numeric vector of size 4. The vertical coordinates of opposite points.

Value
Numeric vector size 2.

Author(s)
Josef Brejcha

References

---

trimeze Coordinates of an object defined by CDF and survival functions

Description
Calculates the coordinates of the object defined matrices C1 and C23. Both matrices are two-row.

Usage
trimeze(C1, C23)

Arguments
C1 numerical probability two-row matrix defining survival line
C23 numerical probability two-row matrix defining CDF line
Value

A list with components as follows:

- **tlc**: upper left corner coordinates
- **brc**: bottom right corner coordinates
- **sp**: survival line coordinates
- **cp**: CDF line coordinates

Author(s)

Josef Brejcha

---

*vfalihaq*  
Ali-Mikhail-Haq Copula Variable Given Second One and Copula Probability

Description

v for Ali-Mikhail-Haq copula \( C(u, v) \) given probability \( C(u, v) \) and \( u \).

Usage

*vfalihaq*(*C*, *u*, *tht*)

Arguments

- **C**: Probability value of the Ali-Mikhail-Haq copula. It can be a vector.
- **u**: The first variable value of the \( C(u, v) \). \( u \) can be a vector if \( C \) is a single. \( u \) is a matrix with \( nrow = length(C) \) if \( C \) is a vector.
- **tht**: Copula parameter

Details

The value of the \( u \) must be greater than \( C \).

Value

The value of the second variable depending on the first variable and copula probability value.

Author(s)

Josef Brejcha
Examples

```r
require(copula)
C = 0.3
tht = 0.5
u = c(0.35, 0.40, 0.45)
v <- vfalihaq(C, u, tht)
kal <- archmCopula(family = "amh", param = tht, dim = 2)
pCopula(cbind(u, v), kali)
```

```
Cf <- c(0.3, 0.4)
mx <- matrix(c(seq(0.35, 0.45, 0.05), seq(0.5, 0.6, 0.05)),
              nrow = 2, ncol = 3, byrow = TRUE)
rownames(mx) <- Cf
vfalihaq(C = Cf, u = mx , tht=0.5)
```

```
[,1]  [,2]  [,3]
0.3  0.8019802 0.6739130 0.6153846
0.4  0.7500000 0.6739130 0.6153846
```

vfclayton

Clayton Copula Variable Given Second One and Copula Probability

Description

`v` for Clayton copula `C(u, v)` given probability `C(u, v)` and `u`.

Usage

```
vfclayton(C, u, tht)
```

Arguments

- **C**: Probability value of the Clayton copula. It can be a vector.
- **u**: The first variable value of the `C(u, v)`. `u` can be a vector if `C` is a single. `u` is a matrix with `nrow = length(C)` if `C` is a vector.
- **tht**: Copula parameter

Details

The value of the `u` must be greater than `C`.

Value

The value of the second variable depending on the first variable and copula probability value.

Author(s)

Josef Brejcha
Examples

C <- 0.3
tht <- 6
u <- c(0.35, 0.4, 0.45)
v <- vfclayton(C, u, tht)
kop = claytonCopula(tht)
pCopula(cbind(u, v), kop)
#
Cf <- c(0.3, 0.4)
mx <- matrix(c(seq(0.35, 0.45, 0.05), seq(0.5, 0.6, 0.05)),
            nrow = 2, ncol = 3, byrow = TRUE)
rownames(mx) <- Cf
vfclayton(C = Cf, u = mx, tht=7)
# [,1]   [,2]   [,3]
# 0.3 0.3183261 0.3061926 0.3025859
# 0.4 0.4135555 0.4064530 0.4033610

vfenuo

Expected values of marginal distributions

Description

Auxiliary function that calculates the expected values of marginal distributions.

Usage

vfenuo(marg, xo)

Arguments

marg  Character vector size greater than or equal to 2. Its components can now be
c("weibull", "gamma", "lnorm", "norm", "betapr", "beta").

xo  Vector size 2*length(marg) of parameters of marg.

    xo[odd]  scale, meanlog, mean, shape1
    xo[even]  shape, sdlog, sd, shape2

Value

Numeric vector size equal to length(marg).

Author(s)

Josef Brejcha
vfex

Examples

vfenuo(marg = c("beta", "norm", "weibull"),
    xo = c(5, 5, 3, 20, 30, 5, 100, 1.5))

vfex

Compute vector V for C(u, V)

Description

A vector \( v \) is computed for \( C \) and numeric probability vector \( u \).

Usage

vfex(C, u, th, fm)

Arguments

- **C**: Copula probability. It is a single value.
- **u**: Probability vector. All its components are greater than \( C \).
- **th**: Copula parameter.

Value

Numeric vector.

Author(s)

Josef Brejcha

vffgm

Farlie-Gumbel-Morgenstern Copula Variable Given Second One and Copula Probability

Description

\( v \) for Farlie-Gumbel-Morgenstern copula \( C(u, v) \) given probability \( C(u, v) \) and \( u \).

Usage

vffgm(C, u, tht)
Arguments

C Probability value of the Farlie-Gumbel-Morgenstern copula. It can be a vector.

u The first variable value of the \( C(u, v) \). \( u \) can be a vector if \( C \) is a single. \( u \) is a matrix with \( \text{nrow} = \text{length}(C) \) if \( C \) is a vector.

tht Copula parameter

Details

The value of the \( u \) must be greater than \( C \).

Value

The value of the second variable depending on the first variable and copula probability value.

Author(s)

Josef Brejcha

References


Examples

```R
require(copula)
C = 0.3
tht = 0.5
u = c(0.35, 0.40, 0.45)
v <- vffgm(C, u, tht)
kfgm <- fgmCopula(tht)
pCopula(c(u, v), kfgm)
# [1] 0.8064052 0.6853009 0.6007056

Cf <- c(0.3, 0.4)
mx <- matrix(c(seq(0.35, 0.45, 0.05), seq(0.5, 0.6, 0.05)),
             nrow = 2, ncol = 3, byrow = TRUE)
rownames(mx) <- Cf
vffgm(C = Cf, u = mx, tht=0.5)
# [,1] [,2] [,3]
# 0.3 0.8064052 0.6853009 0.6007056
# 0.4 0.7535751 0.6781648 0.6195239
```
vffrank

**Frank Copula Variable Given Second One and Copula Probability**

Description

\( v \) for Frank copula \( C(u, v) \) given probability \( C(u, v) \) and \( u \).

Usage

\[
vffrank(C, u, tht)
\]

Arguments

- **C**: Probability value of the Frank copula. It can be a vector.
- **u**: The first variable value of the \( C(u, v) \). \( u \) can be a vector if \( C \) is a single. \( u \) is a matrix with \( nrow = length(C) \) if \( C \) is a vector.
- **tht**: Copula parameter

Details

The value of the \( u \) must be greater than \( C \).

Value

The value of the second variable depending on the first variable and copula probability value.

Author(s)

Josef Brejcha

Examples

\[
C <- 0.3
tht <- 6
u <- c(0.35, 0.4, 0.45)
v <- vffrank(C, u, tht)
kop = frankCopula(tht)
pCopula(cbind(u, v), kop)
\]
Description

v for Gumbel copula C(u, v) given probability C(u, v) and u.

Usage

vfgumbel(C, u, tht)

Arguments

C Probability value of the Gumbel copula. It can be a vector.

u The first variable value of the C(u, v). u can be a vector if C is a single. u is a matrix with nrow = length(C) if C is a vector.

tht Copula parameter

Details

The value of the u must be grater than C.

Value

The value of the second variable depending on the first variable and copula probability value.

Author(s)

Josef Brejcha

Examples

C <- 0.3
tht <- 6
u <- c(0.35, 0.4, 0.45)
v <- vfgumbel(C, u, tht)
kop = gumbelCopula(tht)
pCopula(cbind(u, v), kop)

# vfgumbel(c(0.3, 0.4), u = rbind(seq(0.35, 0.45, 0.05),
# seq(0.45, 0.55, 0.05)), 8)
# [,1] [,2] [,3]
# [1,] 0.3184504 0.3053987 0.3017235
# [2,] 0.4184819 0.4051936 0.4015295
vfjoe

Joe Copula Variable Given Second One and Copula Probability

Description

v for Joe copula C(u, v) given probability C(u, v) and u.

Usage

vfjoe(C, u, tht)

Arguments

- **C**: Probability value of the Joe copula. It can be a vector.
- **u**: The first variable value of the C(u, v). u can be a vector if C is a single. u is a matrix with nrow = length(C) if C is a vector.
- **tht**: Copula parameter

Details

The value of the u must be greater than C.

Value

The value of the second variable depending on the first variable and copula probability value.

Author(s)

Josef Brejcha

Examples

```r
C <- 0.3
tht <- 6
u <- c(0.35, 0.4, 0.45)
v <- vfjoe(C, u, tht)
kop <- joeCopula(tht)
pCopula(cbind(u, v), kop)

Cf <- c(0.3, 0.4)
mx <- matrix(c(seq(0.35, 0.45, 0.05), seq(0.5, 0.6, 0.05)),
            nrow = 2, ncol = 3, byrow = TRUE)
rownames(mx) <- Cf
vfjoe(C = Cf, u = mx, tht=6)
```

```r
# [,1] [,2] [,3]
# [1,] 0.4821216 0.3513741 0.3274672
# [2,] 0.4379531 0.4184746 0.4087143
```


vfmrg  

Auxiliary function

Description

Auxiliary function used in vfploto. It computes random variable value of the CDF or survival which can be one of the c("weibull", "gamma", "lnorm", "norm", "betapr", "beta").

Usage

vfmrg(rdj, i, cosi, yo, cdf)

Arguments

rdj  
A character vector. Its components are from c("weibull", "gamma", "lnorm", "norm", "betapr", "beta").

i  
An index of the rdj

cosi  
A vector of probabilities

yo  
Vector size 2*length(rdj) of parameters of rdj

  yo[1], yo[3]  scale, meanlog, mean, shape1  
  yo[2], yo[4]  shape, sdlog, sd, shape2

cdf  
Cumulative distribution function when TRUE, survival otherwise.

Details

"betapr" is the name of 'BetaPrime' distribution from extrDistr package. The other name 'BetaPrime' is 'Inverted Beta'.

Value

Numeric vector

Author(s)

Josef Brejcha
vfploto

Plotting the cumulative distribution function or survival function

Description

Plotting the cumulative distribution function or survival function.

Usage

vfploto(cx, pro, fam, marg, xo, tht, cdf=TRUE, plt=TRUE, rtn=FALSE, ped = TRUE)

Arguments

cx
A vector of copula probabilities.

pro
Numeric vector. Its pro[1] is upper value of the u. Next pro[-1] are then all greater than or equal to 1. The second case of pro is all pro less than 1. The first case is an extra calculation of the u values. In the latter case, u values can be pre-selected.

fam

marg
A vector size 2. Combination of these marginals: c("weibull", "gamma", "lnorm", "norm", "betapr", "beta").

xo
A vector of marginal distribution parameters. It is size 4 with these components:

xo[1], xo[3] scale, meanlog, mean, shape1
xo[2], xo[4] shape, sdlog, sd, shape2

tht
copula parameter

cdf
logical; Computation for CDF when TRUE. If FALSE is the same for Survival.

plt
Plot only when TRUE.

rtn
Print output value only when TRUE.

ped
Compute and add to plot an expected values of marginal distributions when ped = TRUE.

Details

Must not be plt and rtn at the same time equal to FALSE.
Value

If `rtn` is TRUE, then a list of these components:

- **Type**: character; "CDF" or "Survival"
- **P**: numeric; CDF or Survival value
- **x**: numeric vector of the first marginal values for `P`
- **y**: numeric vector of the second marginal values for `P`
- **u**: numeric vector of the first copula marginal values
- **v**: numeric vector of the second copula marginal values

Author(s)

Josef Brejcha

Examples

```r
require(copula)
tht = 0.475
cx = c(0.0025, 0.05, seq(0.1, 0.9, 0.1), 0.95, 0.975)
# nC = length(cx)
proh = c(0.9999999, 8, 4, 4, 4)
prod = c(0.999, 8, 4, 4, 4)
fam = "clayton"
marg = c("weibull", "lnorm")
ox = c(100, 1.5, 3, 0.425)

suro = vfploto(cx, proh, fam, marg, xo, tht, cdf=FALSE, plt=TRUE, rtn=FALSE)
cdfo = vfploto(cx, prod, fam, marg, xo, tht, cdf=TRUE, plt=TRUE, rtn=FALSE)

##
cx = 0.4
vfploto(cx, proh, fam, marg, xo, tht, cdf=TRUE, plt=FALSE, rtn=TRUE, ped = TRUE)
```

vfprifo

*Computation of the vector `u` to compute the second vector `v`*

Description

Auxiliary function. Each vector value `u` must be greater than the probability of the copula.

Usage

```r
vfprifo(ck, pro)
```

Arguments

- **ck**: Copula probability. Single value. Not a vector.
- **pro**: Numeric vector. All its components are less than 1. `u` can be pre-set in the desired values.
vfpripo

Value
Numeric vector.

Author(s)
Josef Brejcha

Description
Auxiliary function. Each vector value $u$ must be greater than the probability of the copula.

Usage
vfpripo(ck, pro)

Arguments
pro Numeric vector. Its $pro[1:k]$ are upper values of the $u$. Next $pro[-c(1:k)]$ are then all greater than or equal to 1.

Value
Numeric vector.

Author(s)
Josef Brejcha

Examples
prk = c(0.99999, 0.9999, 0.999, 0.99, 8, 4, 2)
C = 0.1
u = vfpripo(ck = C, pro = prk)
Index

gentruk, 3
kopula, 4
prosim, 5
prunikus, 6
trimeze, 3, 7
vfalihaq, 8
vfclayton, 9
vfcp-package, 2
vfenuo, 10
vfex, 11
vffgm, 11
vffrank, 13
vfgumbel, 14
vfjoe, 15
vfmr, 16
vfploto, 17
vfprifo, 18
vfpripo, 19