Package 'bivquant'

August 28, 2019

Title Estimation of Bivariate Quantiles

Version 0.1

Author
Nadja Klein <nadja.klein@hu-berlin.de>, Paul Wiemann <pwiemann@uni-goettingen.de>

Maintainer Nadja Klein <nadja.klein@hu-berlin.de>

Description Computation of bivariate quantiles via linear programming based on a novel direction-based approach using the cumulative distribution function.

Depends R (>= 3.0.1)

Imports cubature, regpro, lpSolve, mvtnorm, copula, MASS

License GPL-2

LazyData true

Collate 'utils.r' 'utils_lp.r' 'bivquant.r' 'dgp.r' 'geomqu.r' 'plot.r'

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-08-28 12:00:02 UTC

R topics documented:

`ball_2d` *On Affine Equivariant Multivariate Quantiles*

Description

This function generates the directions u for which the geometric quantiles are computed.

Usage

```
ball_2d(r, n = 10)
```

Arguments

`r`	length of u computed as $\sqrt{u_1^2 + u_2^2}$.
`n`	number of u's, default is 10.

Details

The Eulicdean norm of u has to be ≤ 1, see Chakraborty, B. (2001).

Value

a matrix of dimension (n,2) containing rowwise the distrinct u's.

Author(s)

Nadja Klein.

References

Chakraborty, B. (2001). On affine equivariant multivariate quantiles. *Annals of the Institute of Statistical Mathematics*, **53**, 380–403. <https://doi.org/10.1023/A:1012478908041>.

See Also

`geomqu`.

bivquant *Bivariate Quantiles*

Description

This function fits the empirical bivariate quantiles based on the CDF (cumulative distrubtion function). We use linear programming. Currently, the solver is lp from the package lpSolve

Usage

```
bivquant(y, alphaseq = NULL, tau = NULL, transformed = FALSE)
```

Arguments

y	the responses in a matrix or data frame with 2 columns and rows equal to the number of observations.
alphaseq	The angles along which the quantile should be computed, can be a vector. If not specified, quantiles will will be computed for a equidistant grid from 0 to $\pi/2$ of length 10 is used.
tau	The quantile level. If not specified, the median, $\tau = 0.5$ will be computed.
transformed	Default is FALSE specifying that quantiles on the original scale are returned. If TRUE, quantiles on the unit square are returned in addition.

Details

The function imitates rotation around (1,1) in the transformed coordinate system and thus allows to estimate the marginal quantiles.

Value

an object of class bivquant.

Author(s)

Nadja Klein.

References

Nadja Klein and Thomas Kneib (2019). Directional Bivariate Quantiles - A Robust Approach based on the Cumulative Distribution Function. To appear in Advances in Statistical Analysis (AStA)

See Also

lp.

Examples

```
require("MASS")
require("mvtnorm")
set.seed(42)

tauseq <- seq(0.1,0.9,by=0.1) #quantile levels

alphas <- seq(0*pi/32,16*pi/32,by=0.5*pi/32) #grid of angles
n <- 50 #sample size

#generate bivariate data
mu <- c(6, 10)
#correlated responses
rho <- 0.5
Sigma <- matrix(c(
    1.0, rho,
    rho, 1.0
  ),
  ncol=2, byrow=TRUE)

X <- rmvnorm(n, mu, Sigma)
bivqu <- bivquant(X,alpha=alphas,tau=tauseq)
plot(bivqu, pch=20,col="grey")


#bigger n
set.seed(123)
n <- 100


X <- dgp_cop(n, family="clayton", margins=c("norm", "norm"),
 paramMargins=list(list(mean = 4, sd = 1), list(mean = 4, sd = 5)),
 rho=1.75)


bivqu <- bivquant(X,alpha=alphas,tau=tauseq)
plot(bivqu, pch=20,col="grey")
```

---

dgp_cop *Generate bivariate data with different margins and dependence structures from Archimedian copulas.*

Description

Generate bivariate data with different margins and dependence structures from Archimedian copulas.

Usage

```
dgp_cop(n, family = "clayton", margins = c("norm", "norm"),
  paramMargins = list(list(mean = 0, sd = 1), list(mean = 0, sd = 1)),
  rho = 2, ...)
```

Arguments

n	sample size.
family	the copula of type Archimedian.
margins	the margins to be specified.
paramMargins	parameters of marginal distributions.
rho	the copula parameter
...	further arguments for function mvdc.

Value

a $nx2$ matrix of the data which is an object of class mvdc.

Author(s)

Nadja Klein.

See Also

rMvdc and mvdc in copula-package for details.

---

dgp_rot — *Generate rotated bivariate normal data compared to a bivariate normal distribution with a certain mean and covariance matrix.*

Description

Generate rotated bivariate normal data compared to a bivariate normal distribution with a certain mean and covariance matrix.

Usage

```
dgp_rot(n, mu, Sigma, gamma = pi/4)
```

Arguments

n	sample size.
mu	expectation, a vector of size two.
Sigma	covariance matrix.
gamma	angle of rotation in radients.

Value

a $n \times 2$ matrix of the data.

Author(s)

Nadja Klein.

`dgp_scale` *Generate scaled bivariate normal data compared to a bivariate normal distribution with unit marginal variances.*

Description

Generate scaled bivariate normal data compared to a bivariate normal distribution with unit marginal variances.

Usage

```
dgp_scale(n, mu, Sigma, v = c(1, 1))
```

Arguments

`n`	sample size.
`mu`	expectation, a vector of size two.
`Sigma`	covariance matrix.
`v`	scaling vector.

Value

a $n \times 2$ matrix of the data.

Author(s)

Nadja Klein.

`dgp_shear` *Generate sheared bivariate normal data compared to a bivariate normal distribution with a certain mean and covariance matrix.*

Description

Generate sheared bivariate normal data compared to a bivariate normal distribution with a certain mean and covariance matrix.

Usage

```
dgp_shear(n, mu, Sigma, v = c(1, 1))
```

Arguments

`n`	sample size.
`mu`	expectation, a vector of size two.
`Sigma`	covariance matrix.
`v`	shearing vector.

Value

a $n \times 2$ matrix of the data.

Author(s)

Nadja Klein.

`dgp_trans` *Generate translated bivariate normal data (with a certain mean and covariance matrix) compared to a zero mean distribution.*

Description

Generate translated bivariate normal data (with a certain mean and covariance matrix) compared to a zero mean distribution.

Usage

```
dgp_trans(n, mu, Sigma, v = c(0, 0))
```

Arguments

n	sample size.
mu	expectation, a vector of size two.
Sigma	covariance matrix.
v	translation vector.

Value

a $n \times 2$ matrix of the data.

Author(s)

Nadja Klein.

---

find_good_alpha *On Affine Equivariant Multivariate Quantiles*

Description

This function returns the transformation matrix X(a), see Chakraborty, B. (2001).

Usage

```
find_good_alpha(X, eps = 0.025)
```

Arguments

X	must be a (n,2) matrix.
eps	stopping criterion.

Value

a list with a the selected rows of the data, X is the X(a) matrix, X_reduced is the data matrix X without the selected columns.

Author(s)

Nadja Klein, Paul Wiemann.

References

Chakraborty, B. (2001). On affine equivariant multivariate quantiles. *Annals of the Institute of Statistical Mathematics*, **53**, 380–403. <https://doi.org/10.1023/A:1012478908041>.

See Also

geomqu.

geomqu *On Affine Equivariant Multivariate Quantiles*

Description

This function fits empirical bivariate quantiles as proposed by Chakraborty, B. (2001) <https://doi.org/10.1023/A:1012478908

Usage

```
geomqu(data, u, p, alpha, keep_optim = FALSE, keep_data = TRUE)
```

Arguments

data	must be a (n,d) matrix of observations.
u	a (m,d) matrix with m directions.
p	defines the p-norm, must be in $[1, \infty)$.
alpha	must be missing or a vector of length d+1 with distict values from 1 to n.
keep_optim	boolean keep results from optimization in result object, default is FALSE.
keep_data	boolean keep data parameter in result object, default is TRUE.

Value

an object of class geomqu with methods plot.geomqu and plot.geomqu.

Author(s)

Nadja Klein.

References

Chakraborty, B. (2001). On affine equivariant multivariate quantiles. *Annals of the Institute of Statistical Mathematics*, **53**, 380–. <https://doi.org/10.1023/A:1012478908041>.

geomqu2d_norm2 *On Affine Equivariant Multivariate Quantiles*

Description

This function fits empirical bivariate quantiles as proposed by Chakraborty, B. (2001). 2 dimensional data and with p=2

Usage

```
geomqu2d_norm2(data, probs, alpha, k = 8)
```

Arguments

data	must be a (n,2) matrix of observations.
probs	vector of probs which is used to calculate the us
alpha	missing or a vector of length 3 with distict values form 1 to n
k	number of us per prob

Author(s)

Nadja Klein.

References

Chakraborty, B. (2001). On affine equivariant multivariate quantiles. *Annals of the Institute of Statistical Mathematics*, **53**, 380–403.

See Also

geomqu for details.

Examples

```
require("MASS")
require("mvtnorm")
set.seed(42)
n <- 50
mu <- c(6, 10)
#correlated responses
rho <- 0.5
Sigma <- matrix(c(
    1.0, rho,
    rho, 1.0
  ),
  ncol=2, byrow=TRUE)

X <- rmvnorm(n, mu, Sigma)
result <- geomqu2d_norm2(X, probs=c(0.8,0.9), k=8)
plot(result)

#now independent responses
rho <- 0.0
Sigma <- matrix(c(
    1.0, rho,
    rho, 1.0
  ),
  ncol=2, byrow=TRUE)

X <- rmvnorm(n, mu, Sigma)
result <- geomqu2d_norm2(X, probs=c(0.8,0.9), k=8)
plot(result)
```

```
#now some non-normal data
X <- dgp_cop(n, family="clayton", margins=c("norm", "norm"),
 paramMargins=list(list(mean = 4, sd = 1), list(mean = 4, sd = 5)),
 rho=1.75)
result <- geomqu2d_norm2(X, probs=c(0.8,0.9), k=8)
plot(result)
```

`plot.bivquant` *Plot the estimated bivariate quantiles (for the CDF (cumulative distrubtion function)-based quantiles) and the data for a fitted bivquant object. See ?bivquant for more details.*

Description

Plot the estimated bivariate quantiles (for the CDF (cumulative distrubtion function)-based quantiles) and the data for a fitted bivquant object. See ?bivquant for more details.

Usage

```
## S3 method for class 'bivquant'
plot(x,
  mains = c("CDF-based quantiles (original scale)",
  "CDF-based quantile (unit square)"), ...)
```

Arguments

`x`	a fitted `bivquant` model.
`mains`	title of figures.
`...`	further arguments for function `plot`.

Author(s)

Nadja Klein.

See Also

`plot` for details.

`plot.geomqu` *Plot the estimated bivariate quantiles (for the geometric quantiles of Chakraborty, B. (2001).) and the data. See ?geomqu for more details.*

Description

Plot the estimated bivariate quantiles (for the geometric quantiles of Chakraborty, B. (2001).) and the data. See ?geomqu for more details.

Usage

```
## S3 method for class 'geomqu'
plot(x, ...)
```

Arguments

`x`	a fitted `geomqu` model.
`...`	further arguments for function `plot`.

Author(s)

Nadja Klein, Paul Wiemann.

See Also

`plot` for details.

Index