Package ‘pbv’

November 30, 2023

Type Package

Title Probabilities for Bivariate Normal Distribution

Version 0.5-47

Date 2023-11-30 15:01:12.000296

Author Alexander Robitzsch [aut,cre] (<https://orcid.org/0000-0002-8226-3132>)

Maintainer Alexander Robitzsch <robitzsch@ipn.uni-kiel.de>

Description Computes probabilities of the bivariate normal distribution in a vectorized R function (Drezner & Wesolowsky, 1990, <doi:10.1080/00949659008811236>).

Depends R (>= 3.1)

Imports Rcpp

Enhances pbivnorm

LinkingTo Rcpp, RcppArmadillo

URL https://github.com/alexanderrobitzsch/pbv,
https://sites.google.com/view/alexander-robitzsch/software

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-11-30 15:20:02 UTC

R topics documented:

pbv-package *Probabilities for Bivariate Normal Distribution*

Description

Computes probabilities of the bivariate normal distribution in a vectorized R function (Drezner & Wesolowsky, 1990, <doi:10.1080/00949659008811236>).

Author(s)

Alexander Robitzsch [aut,cre] (<https://orcid.org/0000-0002-8226-3132>)

Maintainer: Alexander Robitzsch <robitzsch@ipn.uni-kiel.de>

References

Drezner, Z., & Wesolowsky, G. O. (1990). On the computation of the bivariate normal integral. *Journal of Statistical Computation and Simulation, 35*(1-2), 101-107. doi:10.1080/00949659008811236

pbvnorm *Probabilities for Bivariate Normal Distribution*

Description

The function `pbvnorm` computes probabilities $\Phi_2(x, y, \rho)$ for the standardized bivariate normal distribution (Drezner & Wesolowsky, 1990; West, 2004).

The function `dbvnorm` computes the corresponding density $\phi_2(x, y, \rho)$.

Usage

```
pbvnorm(x, y, rho)

dbvnorm(x, y, rho, log=FALSE)

## exported Rcpp functions
pbv_rcpp_pbvnorm0( h1, hk, r)
pbv_rcpp_pbvnorm( x, y, rho)
pbv_rcpp_dbvnorm0( x, y, rho, use_log)
pbv_rcpp_dbvnorm( x, y, rho, use_log)
```

Arguments

`x`	Vector of first ordinate
`y`	Vector of second ordinate
`rho`	Vector of correlations
`log`	Logical indicating whether logarithm of the density should be calculated
`h1`	Numeric
`hk`	Numeric
`r`	Numeric
`use_log`	Logical

Value

A vector

Note

The **pbv** package can also be used to include **Rcpp** functions for computing bivariate probabilities at the C++ level. Numeric and vector versions are

`double pbv::pbv_rcpp_pbvnorm0( double h1, double hk, double r)`

`Rcpp::NumericVector pbv::pbv_rcpp_pbvnorm( Rcpp::NumericVector x, Rcpp::NumericVector y, Rcpp::NumericVector rho)`

References

Drezner, Z., & Wesolowsky, G. O. (1990). On the computation of the bivariate normal integral. *Journal of Statistical Computation and Simulation, 35*(1-2), 101-107. doi:10.1080/00949659008811236

Genz, A. (1992). Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics, 1*(2), 141-149.

West, G. (2005). Better approximations to cumulative normal functions. *Wilmott Magazine, 9*, 70-76.

See Also

See `pbivnorm::pbivnorm` in the **pbivnorm** package and `mnormt::biv.nt.prob` in the **mnormt** package for alternative implementations (Genz, 1992).

Examples

```
#############################################################################
# EXAMPLE 1: Comparison with alternative implementations
#############################################################################

#*** simulate different values of ordinates and correlations
set.seed(9898)
N <- 3000
```

```
x <- stats::runif(N,-3,3)
y <- stats::runif(N,-3,3)
rho <- stats::runif(N,-.95,.95)

#*** compute probabilities
res1 <- pbv::pbvnorm(x=x,y=y,rho=rho)

## Not run:
#-- compare results with pbivnorm package
library(pbivnorm)
res2 <- pbivnorm::pbivnorm(x=x, y=y, rho=rho)

summary(abs(res1-res2))

#*** compute density values
log <- TRUE    # logical indicating whether log density should be evaluated
res1 <- pbv::dbvnorm(x=x, y=y, rho=rho, log=log )

#-- compare results with mvtnorm package
library(mvtnorm)
res2 <- rep(NA, N)
sigma <- diag(2)
for (ii in 1:N){
    sigma[1,2] <- sigma[2,1] <- rho[ii]
    res2[ii] <- mvtnorm::dmvnorm(x=c(x[ii],y[ii]), sigma=sigma, log=log)
}
summary(abs(res1-res2))

## End(Not run)
```

Index