Package ‘TruncatedNormal’
August 31, 2019

Type Package
Title Truncated Multivariate Normal and Student Distributions
Version 2.1
Date 2019-08-30
License GPL-3

BugReports https://github.com/lbelzile/TruncatedNormal/issues
Imports nleqslv, randtoolbox, Rcpp (>= 0.12.16)
LinkingTo Rcpp, RcppArmadillo
RoxygenNote 6.1.1
VignetteBuilder knitr
Encoding UTF-8
Suggests knitr, rmarkdown, mvtnorm
NeedsCompilation yes

Author Zdravko Botev [aut] (<https://orcid.org/0000-0001-9054-3452>),
Leo Belzile [aut, cre] (<https://orcid.org/0000-0002-9135-014X>)
Maintainer Leo Belzile <belzilel@gmail.com>
Repository CRAN
Date/Publication 2019-08-31 05:00:02 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>TruncatedNormal-package</td>
<td>2</td>
</tr>
<tr>
<td>cholperm</td>
<td>2</td>
</tr>
<tr>
<td>lupus</td>
<td>3</td>
</tr>
<tr>
<td>pmvnorm</td>
<td>4</td>
</tr>
<tr>
<td>pmvt</td>
<td>5</td>
</tr>
<tr>
<td>tmvnorm</td>
<td>6</td>
</tr>
</tbody>
</table>
TruncatedNormal-package

Truncated Normal Distribution Toolbox

Description

The routines include:

• generator of independent and identically distributed random vectors from the truncated univariate and multivariate distributions;
• (Quasi-) Monte Carlo estimator and a deterministic upper bound of the cumulative distribution function of the multivariate normal and Student distributions;
• algorithm for the accurate computation of the quantile function of the normal distribution in the extremes of its tails.

Author(s)

Leo Belzile and Z. I. Botev, email: <botev@unsw.edu.au> and web page: https://web.maths.unsw.edu.au/~zdravkobotev/

References

• Z. I. Botev and P. L’Ecuyer (2015), Efficient Estimation and Simulation of the Truncated Multivariate Student-t Distribution, Proceedings of the 2015 Winter Simulation Conference, Huntington Beach, CA, USA

cholperm

Cholesky decomposition for Gaussian distribution function with permutation

Description

This function computes the Cholesky decomposition of a covariance matrix Sigma and returns a list containing the permuted bounds for integration. The prioritization of the variables follows either the rule proposed in Gibson, Glasbey and Elston (1994), reorder variables to have outermost variables with smallest expected values. The alternative is the scheme proposed in Genz and Bretz (2009) that minimizes the variance of the truncated Normal variates.
Usage

cholperm(Sigma, l, u, method = c("GGE", "GB"))

Arguments

Sigma d by d covariance matrix
l d vector of lower bounds
u d vector of upper bounds
method string indicating which method to use. Default to "GGE"

Details

The list contains an integer vector perm with the indices of the permutation, which is such that Sigma(perm,perm) == L %*% t(L). The permutation scheme is described in Genz and Bretz (2009) in Section 4.1.3, p.37.

Value

a list with components

- L: Cholesky root
- l: permuted vector of lower bounds
- u: permuted vector of upper bounds
- perm: vector of integers with ordering of permutation

References


---

lupus

Latent membranous Lupus Nephritis dataset

Description

The data represents two clinical measurements (covariates), which are used to predict the occurrence of latent membranous lupus nephritis. The dataset consists of measurements on 55 patients of which 18 have been diagnosed with latent membranous lupus.

Usage

data("lupus")
pmvnorm

Format
A data frame with columns "response", "const", "x1" and "x2"

Details
The data were transcribed from Table 1, page 22, of Dyk and Meng (2001).

References

See Also
mvrandn, which uses this dataset.

Examples
data("lupus")

pmvnorm

Distribution function of the multivariate normal distribution for arbitrary limits

Description
This function computes the distribution function of a multivariate normal distribution vector for an arbitrary rectangular region \([lb, ub]\). pmvnorm computes an estimate and the value is returned along with a relative error and a deterministic upper bound of the distribution function of the multivariate normal distribution. Infinite values for vectors \(u\) and \(l\) are accepted. The Monte Carlo method uses sample size \(n\): the larger the sample size, the smaller the relative error of the estimator.

Usage
pmvnorm(mu, sigma, lb = -Inf, ub = Inf, B = 10000, type = c("mc", "qmc"), log = FALSE)

Arguments
mu vector of location parameters
sigma covariance matrix
lb vector of lower truncation limits
ub vector of upper truncation limits
B number of replications for the (quasi)-Monte Carlo scheme
type string, either of mc or qmc for Monte Carlo and quasi Monte Carlo, respectively
log logical; if TRUE, probabilities and density are given on the log scale.
pmvt

Author(s)
Zdravko I. Botev, Leo Belzile (wrappers)

References

See Also
pmvnorm

Examples

#From mvtnorm
mean <- rep(0, 5)
lower <- rep(-1, 5)
upper <- rep(3, 5)
corr <- matrix(0.5, 5, 5) + diag(0.5, 5)
prob <- pmvnorm(lb = lower, ub = upper, mu = mean, sigma = corr)
stopifnot(pmvnorm(lb = -Inf, ub = 3, mu = 0, sigma = 1) == pnorm(3))

pmvt

Distribution function of the multivariate Student distribution for arbitrary limits

Description
This function computes the distribution function of a multivariate normal distribution vector for an arbitrary rectangular region \([lb, ub]\). pmvt computes an estimate and the value is returned along with a relative error and a deterministic upper bound of the distribution function of the multivariate normal distribution. Infinite values for vectors \(u\) and \(l\) are accepted. The Monte Carlo method uses sample size \(n\): the larger the sample size, the smaller the relative error of the estimator.

Usage

pmvt(mu, sigma, df, lb = -Inf, ub = Inf, type = c("mc", "qmc"), 
log = FALSE, B = 10000)

Arguments

mu vector of location parameters
sigma scale matrix
df degrees of freedom
lb vector of lower truncation limits
ub vector of upper truncation limits
tmvnorm

Multivariate truncated normal distribution

Description

Density, distribution function and random generation for the multivariate truncated normal distribution with mean vector mu, covariance matrix sigma, lower truncation limit lb and upper truncation limit ub. The truncation limits can include infinite values. The Monte Carlo (type = "mc") uses a sample of size B, while the quasi Monte Carlo (type = "qmc") uses a pointset of size ceiling(n/12) and estimates the relative error using 12 independent randomized QMC estimators.

Arguments

- type: string, either of mc or qmc for Monte Carlo and quasi Monte Carlo, respectively
- log: logical; if TRUE, probabilities and density are given on the log scale.
- B: number of replications for the (quasi)-Monte Carlo scheme

Author(s)

Matlab code by Zdravko I. Botev, R port by Leo Belzile

References


Examples

d <- 15; nu <- 30;
l <- rep(2, d); u <- rep(Inf, d);
sigma <- 0.5 * matrix(1, d, d) + 0.5 * diag(1, d);
est <- pmvt(lb = l, ub = u, sigma = sigma, df = nu)

# mvtnorm::pmvt(lower = l, upper = u, df = nu, sigma = sigma)

## Not run:
d <- 5
sigma <- solve(0.5 * diag(d) + matrix(0.5, d, d))
# mvtnorm::pmvt(lower = rep(-1,d), upper = rep(Inf, d), df = 10, sigma = sigma)[1]

pmvt(lb = rep(-1, d), ub = rep(Inf, d), sigma = sigma, df = 10)

## End(Not run)
**tmvnorm**

sigma  
covariance matrix

lb  
vector of lower truncation limits

ub  
vector of upper truncation limits

type  
string, either of mc or qmc for Monte Carlo and quasi Monte Carlo, respectively

**Value**

dtmvnorm gives the density, ptmvnorm and pmvnorm give the distribution function of respectively the truncated and multivariate Gaussian distribution and rtmvnorm generate random deviates.

**Usage**

dtmvnorm(x, mu, sigma, lb, ub, type = c("mc", "qmc"), log = FALSE, B = 1e4)
ptmvnorm(q, mu, sigma, lb, ub, type = c("mc", "qmc"), log = FALSE, B = 1e4)
rtmvnorm(n, mu, sigma, lb, ub)

**Author(s)**

Zdravko I. Botev, Leo Belzile (wrappers)

**References**


**Examples**

d <- 4; lb <- rep(0, d)
mu <- runif(d)
sigma <- matrix(0.5, d, d) + diag(0.5, d)
samp <- rtmvnorm(n = 10, mu = mu, sigma = sigma, lb = lb)
loglik <- dtmvnorm(samp, mu = mu, sigma = sigma, lb = lb, log = TRUE)
cdf <- ptmvnorm(samp, mu = mu, sigma = sigma, lb = lb, log = TRUE, type = "q")

# Exact Bayesian Posterior Simulation Example
# Vignette, example 5
## Not run:
data("lupus"); # load lupus data
Y <- lupus[,1]; # response data
X <- as.matrix(lupus[,-1]) # construct design matrix
n <- nrow(X)
d <- ncol(X)
X <- diag(2*Y-1) %*% X; # incorporate response into design matrix
musq <- 10000; # prior scale parameter
C <- solve(diag(d)/musq + crossprod(X))
sigma <- diag(n) + musq*tcrossprod(X) # this is covariance of Z given beta
est <- pmvnorm(sigma = sigma, lb = 0)
# estimate acceptance probability of crude Monte Carlo
print(attributes(est)$upbnd/est[1])
# reciprocal of acceptance probability
Z <- rtmvnorm(sigma = sigma, n = 1e3, lb = rep(0, n))
# sample exactly from auxiliary distribution
beta <- rtmvnorm(n = nrow(Z), sigma = C) + Z %*% X %*% C
# simulate beta given Z and plot boxplots of marginals
boxplot(beta, ylab = expression(beta))
# output the posterior means
colMeans(beta)

## End(Not run)

tmvt

Multivariate truncated Student distribution

Description
Density, distribution function and random generation for the multivariate truncated Student distribution with location vector \( \mu \), scale matrix \( \sigma \), lower truncation limit \( \ell_b \), upper truncation limit \( \ell_u \) and degrees of freedom \( \nu \).

Arguments
- \( n \): number of observations
- \( x, q \): vector or matrix of quantiles
- \( B \): number of replications for the (quasi)-Monte Carlo scheme
- \( \log \): logical; if TRUE, probabilities and density are given on the log scale.
- \( \mu \): vector of location parameters
- \( \sigma \): scale matrix
- \( \nu \): degrees of freedom
- \( \ell_b \): vector of lower truncation limits
- \( \ell_u \): vector of upper truncation limits
- \( type \): string, either of "mc" or "qmc" for Monte Carlo and quasi Monte Carlo, respectively

Details
The truncation limits can include infinite values. The Monte Carlo (\( type = "mc" \)) uses a sample of size \( B \), while the quasi Monte Carlo (\( type = "qmc" \)) uses a pointset of size \( \text{ceiling}(n/12) \) and estimates the relative error using 12 independent randomized QMC estimators.

pmvt computes an estimate and a deterministic upper bound of the distribution function of the multivariate normal distribution. Infinite values for vectors \( u \) and \( l \) are accepted. The Monte Carlo method uses sample size \( n \): the larger \( n \), the smaller the relative error of the estimator.

Value
dtmvt gives the density, ptmvt gives the distribution function, rtmvt generate random deviates.
```r
Usage
dtmvt(x, mu, sigma, df, lb, ub, type = c("mc", "qmc"), log = FALSE, B = 1e4)
ptmvt(q, mu, sigma, df, lb, ub, type = c("mc", "qmc"), log = FALSE, B = 1e4)
rtmvt(n, mu, sigma, df, lb, ub)
pmvt(mu, sigma, df, lb = -Inf, ub = Inf, type = c("mc", "qmc"), log = FALSE, B = 1e4)
```

**Author(s)**
Leo Belzile, R port from Matlab code by Z. I. Botev

**References**

**Examples**
```r
d <- 4; lb <- rep(0, d)
mu <- runif(d)
sigma <- matrix(0.5, d, d) + diag(0.5, d)
samp <- rtmvt(n = 10, mu = mu, sigma = sigma, df = 2, lb = lb)
loglik <- dtmvt(samp, mu = mu, sigma = sigma, df = 2, lb = lb, log = TRUE)
cdf <- ptmvt(samp, mu = mu, sigma = sigma, df = 2, lb = lb, log = TRUE, type = "q")
```

---

**tnorm**

*Truncated univariate normal distribution*

**Description**
The function provides efficient state-of-the-art random number generation of a vector of truncated univariate distribution of the same length as the lower bound vector. The function is vectorized and the vector of means `mu` and of standard deviations `sd` are recycled.

If `mu` or `sd` are not specified they assume the default values of 0 and 1, respectively.

**Arguments**
- `n`: number of observations
- `p`: vector or matrix of probabilities
- `mu`: vector of means
- `sd`: vector of standard deviations
- `lb`: vector of lower truncation limits
- `ub`: vector of upper truncation limits
- `method`: string, either of fast or invtransfo
Value

vector or matrix of random variates (rtnorm) or of quantiles (ptnorm), depending on the input

Examples

rtnorm(n = 10, mu = 2, lb = 1:10, ub = 2:11, method = "fast")
qtnorm(runif(10), mu = 2, lb = 1:10, ub = 2:11, sd = 1)

Description

Simulates \( n \) random vectors \( X \) exactly distributed from the \( d \)-dimensional Student distribution with \( df = \nu \) degrees of freedom, mean zero and scale matrix \( \sigma \), conditional on \( l < X < u \).

Usage

tregress(n, lb, ub, sigma, df)

Arguments

n number of observations
lb vector of lower truncation limits
ub vector of upper truncation limits
sigma scale matrix
df degrees of freedom

Value

list with components
- \( R \): \( n \) vector of scale
- \( Z \): a \( d \) by \( n \) matrix

so that \( \sqrt{\nu}Z/R \) follows a truncated Student distribution

Author(s)

Matlab code by Zdravko Botev, R port by Leo Belzile

References

Examples

d <- 5
tregress(lb = rep(-2, d), ub = rep(2, d), df = 3, n = 10,
        sigma = diag(0.5, d) + matrix(1, d, d))
Index

*Topic **datasets**
  lupus, 3
*Topic **lupus**
  lupus, 3

cholperm, 2
lupus, 3
mvrandn, 4
pmvnorm, 4, 5
pmvt, 5
tmvnorm, 6
tmvt, 8
tnorm, 9
tregress, 10
TruncatedNormal
  (TruncatedNormal-package), 2
TruncatedNormal-package, 2