Package: TruncatedNormal (via r-universe)

October 9, 2026

Type Package

Title Truncated Multivariate Normal and Student Distributions

Version 2.3

Description A collection of functions to deal with the truncated univariate and multivariate normal and Student distributions, described in Botev (2017) <doi:10.1111/rssb.12162> and Botev and L'Ecuyer (2015) <doi:10.1109/WSC.2015.7408180>.

License GPL-3

BugReports https://github.com/lbelzile/TruncatedNormal/issues

Depends R (>= 2.10)

Imports nleqslv, qrng, spacefillr, alabama, Rcpp (>= 0.12.16)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.3.2

VignetteBuilder knitr

Encoding UTF-8

Suggests knitr, rmarkdown, mvtnorm, carData, tinytest

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 https://cran.r-universe.dev

Date/Publication 2024-07-09 02:44:50 UTC

RemoteUrl https://github.com/cran/TruncatedNormal

RemoteRef HEAD

RemoteSha 4ca8c311452e2740015b38894815b933bd205abe

Contents

TruncatedNormal-package . . . 2
cholperm . . . 3
lupus . . . 4
mroz . . . 4
pmvnorm . . . 5
pmvt . . . 6
tmvnorm . . . 8
tmvt . . . 9
tnorm . . . 11
tregress . . . 11

Index **13**

`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/~zdravkobo

References

- Z. I. Botev (2017), *The Normal Law Under Linear Restrictions: Simulation and Estimation via Minimax Tilting*, Journal of the Royal Statistical Society, Series B, **79** (1), pp. 1–24.
- 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
- Gibson G. J., Glasbey C. A., Elston D. A. (1994), *Monte Carlo evaluation of multivariate normal integrals and sensitivity to variate ordering*, In: Advances in Numerical Methods and Applications, pages 120–126

`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

Genz, A. and Bretz, F. (2009). Computations of Multivariate Normal and t Probabilities, volume 105. Springer, Dordrecht.

Gibson G.J., Glasbey C.A. and D.A. Elton (1994). Monte Carlo evaluation of multivariate normal integrals and sensitivity to variate ordering. In: Dimon et al., Advances in Numerical Methods and Applications, WSP, pp. 120-126.

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.

Format

a data frame with columns "response", "const", "x1" and "x2"

Source

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

References

D. A. van Dyk and X.-L. Meng (2001) *The art of data augmentation (with discussion)*. Journal of Computational and Graphical Statistics, volume 10, pages 1-50.

See Also

The dataset is used in the examples of `mvrandn`

mroz *Women wage dataset from Mroz (1987)*

Description

The data are from the Panel Study of Income Dynamics (PSID) longitudinal study, 1976 wave. They give the number of work hours of married women along with socio-economic variables and the number of children.

Format

a data frame containing the following variables:

- `whrs`: hours of work
- `kidslt6`: number of children aged 5 and below years old in household
- `kidsge6`: number of children between age of 6 and 18 in household
- `age`: age (in years)
- `educ`: number of years in school
- `hearn`: hourly earnings
- `exp`: years of previous labor market experience

Source

W. Greene's website, accessed 17.12.2019 at <http://www.stern.nyu.edu/~wgreene/Text/Edition7/TableF5-1.csv>.

References

T. A. Mroz, 1987. *The Sensitivity of an Empirical Model of Married Women's Hours of Work to Economic and Statistical Assumptions*, Econometrica, **55**(4), pp. 765-799

See Also

Mroz

`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"),
  check = TRUE,
  ...
)
```

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
check	logical; if TRUE (default), the code checks that the scale matrix sigma is positive definite and symmetric
...	additional arguments, currently ignored.

Author(s)

Zdravko I. Botev, Leo Belzile (wrappers)

References

Z. I. Botev (2017), *The normal law under linear restrictions: simulation and estimation via minimax tilting*, Journal of the Royal Statistical Society, Series B, **79** (1), pp. 1–24.

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"),
  B = 10000,
  check = TRUE,
  ...
)
```

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
type	string, either of mc or qmc for Monte Carlo and quasi Monte Carlo, respectively
B	number of replications for the (quasi)-Monte Carlo scheme
check	logical, if TRUE (default), check that the scale matrix sigma is positive definite and symmetric.
...	additional arguments, currently ignored.

Author(s)

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

References

Z. I. Botev and P. L'Ecuyer (2015), Efficient probability estimation and simulation of the truncated multivariate Student-t distribution, Proceedings of the 2015 Winter Simulation Conference, pp. 380-391

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 *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

n	number of observations
x, q	vector 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	covariance matrix
lb	vector of lower truncation limits
ub	vector of upper truncation limits
check	logical; if TRUE (default), the code checks that the scale matrix sigma is positive definite and symmetric
...	additional arguments, currently ignored
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

Z. I. Botev (2017), *The normal law under linear restrictions: simulation and estimation via minimax tilting*, Journal of the Royal Statistical Society, Series B, **79** (1), pp. 1–24.

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
nusq <- 10000; # prior scale parameter
C <- solve(diag(d)/nusq + crossprod(X))
sigma <- diag(n) + nusq*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 `lb`, upper truncation limit `ub` and degrees of freedom `df`.

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
df	degrees of freedom
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
check	logical, if TRUE (default), check that the scale matrix sigma is positive definite and symmetric.
...	additional arguments, currently ignored

Details

The truncation limits can include infinite values. The Monte Carlo (type = "mc") uses a sample of size B, while the qausi Monte Carlo (type = "qmc") uses a pointset of size 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.

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"), B = 1e4)
```

Author(s)

Leo Belzile, R port from Matlab code by Z. I. Botev

References

Z. I. Botev and P. L'Ecuyer (2015), Efficient probability estimation and simulation of the truncated multivariate Student-t distribution, Proceedings of the 2015 Winter Simulation Conference, pp. 380-391

Examples

```
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)
```

tregress *Truncated student generator for Bayesian regression simulation*

Description

Simulates `n` random vectors X exactly distributed from the `d`-dimensional Student distribution with `df`=ν 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

Z. I. Botev and P. L'Ecuyer (2015), Efficient probability estimation and simulation of the truncated multivariate Student-t distribution, Proceedings of the 2015 Winter Simulation Conference, pp. 380-391,

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

cholperm, 3

lupus, 4

Mroz, *5*
mroz, 4
mvrandn, *4*

pmvnorm, 5, *6*
pmvt, 6

tmvnorm, 8
tmvt, 9
tnorm, 11
tregress, 11
TruncatedNormal
        *(*TruncatedNormal-package*)*, 2
TruncatedNormal-package, 2