

# Package ‘mvtnorm’

October 27, 2009

**Title** Multivariate Normal and t Distributions

**Version** 0.9-8

**Date** 2009-05-22

**Author** Alan Genz, Frank Bretz, Tetsuhisa Miwa, Xuefei Mi, Friedrich Leisch, Fabian Scheipl,  
Torsten Hothorn

**Maintainer** Torsten Hothorn <Torsten.Hothorn@R-project.org>

**Description** Computes multivariate normal and t probabilities, quantiles, random deviates and densities.

**Imports** stats

**Depends** R(>= 1.9.0)

**License** GPL-2

**Repository** CRAN

**Date/Publication** 2009-10-27 19:17:43

## R topics documented:

algorithms . . . . .	2
Mvnorm . . . . .	3
Mvt . . . . .	4
pmvnorm . . . . .	5
pmvt . . . . .	7
qmvnorm . . . . .	10
qmvt . . . . .	11
<b>Index</b>	<b>13</b>

**Description**

Choose between two algorithms for evaluating normal distributions and define hyper parameters.

**Usage**

```
GenzBretz(maxpts = 25000, abseps = 0.001, releps = 0)
Miwa(steps = 128)
```

**Arguments**

maxpts	maximum number of function values as integer.
abseps	absolute error tolerance as double.
releps	relative error tolerance as double.
steps	number of grid points to be evaluated.

**Details**

There are two algorithms available for evaluating normal probabilities: The default is the randomized Quasi-Monte-Carlo procedure by Genz (1992, 1993) and Genz and Bretz (2002) applicable to arbitrary covariance structures and dimensions up to 1000.

For smaller dimensions (up to 20) and non-singular covariance matrices, the algorithm by Miwa et al. (2003) can be used as well.

**Value**

An object of class `GenzBretz` or `Miwa` defining hyper parameters.

**References**

- Genz, A. (1992). Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, **1**, 141–150.
- Genz, A. (1993). Comparison of methods for the computation of multivariate normal probabilities. *Computing Science and Statistics*, **25**, 400–405.
- Genz, A. and Bretz, F. (2002), Methods for the computation of multivariate t-probabilities. *Journal of Computational and Graphical Statistics*, **11**, 950–971.
- Genz, A. and Bretz, F. (2009), *Computation of Multivariate Normal and t Probabilities*. Lecture Notes in Statistics, Vol. 195. Springer-Verlag, Heidelberg.
- Miwa, A., Hayter J. and Kuriki, S. (2003). The evaluation of general non-centred orthant probabilities. *Journal of the Royal Statistical Society, Ser. B*, **65**, 223–234.

**Description**

These functions provide the density function and a random number generator for the multivariate normal distribution with mean equal to `mean` and covariance matrix `sigma`.

**Usage**

```
dmvnorm(x, mean, sigma, log=FALSE)
rmvnorm(n, mean = rep(0, nrow(sigma)), sigma = diag(length(mean)),
        method=c("eigen", "svd", "chol"))
```

**Arguments**

<code>x</code>	Vector or matrix of quantiles. If <code>x</code> is a matrix, each row is taken to be a quantile.
<code>n</code>	Number of observations.
<code>mean</code>	Mean vector, default is <code>rep(0, ncol(x))</code> .
<code>sigma</code>	Covariance matrix, default is <code>diag(ncol(x))</code> .
<code>log</code>	Logical; if TRUE, densities <code>d</code> are given as <code>log(d)</code> .
<code>method</code>	Matrix decomposition used to determine the matrix root of <code>sigma</code> , possible methods are eigenvalue decomposition ("eigen", default), singular value decomposition ("svd"), and Cholesky decomposition ("chol").

**Author(s)**

Friedrich Leisch and Fabian Scheipl

**See Also**

[pmvnorm](#), [rnorm](#), [qmvnorm](#)

**Examples**

```
dmvnorm(x=c(0,0))
dmvnorm(x=c(0,0), mean=c(1,1))

sigma <- matrix(c(4,2,2,3), ncol=2)
x <- rmvnorm(n=500, mean=c(1,2), sigma=sigma)
colMeans(x)
var(x)

x <- rmvnorm(n=500, mean=c(1,2), sigma=sigma, method="chol")
colMeans(x)
var(x)
```

```
plot(x)
```

---

Mvt

*The Multivariate t Distribution*

---

### Description

These functions provide information about the multivariate t distribution with non-centrality parameter `delta`, correlation matrix `sigma` and degrees of freedom `df`. `dmvt` gives the density and `rmvt` generates random deviates.

### Usage

```
rmvt(n, sigma = diag(2), df = 1)
dmvt(x, delta, sigma, df = 1, log = TRUE)
```

### Arguments

<code>x</code>	Vector or matrix of quantiles. If <code>x</code> is a matrix, each row is taken to be a quantile.
<code>n</code>	Number of observations.
<code>delta</code>	the vector of noncentrality parameters of length <code>n</code> .
<code>sigma</code>	Correlation matrix, default is <code>diag(ncol(x))</code> .
<code>df</code>	degree of freedom as integer.
<code>log</code>	Logical; if TRUE, densities <code>d</code> are given as <code>log(d)</code> .

### See Also

[pmvt](#) and [qmvt](#)

### Examples

```
dmvt(x=c(0,0), sigma = diag(2))
x <- rmvt(n=100, sigma = diag(2), df = 3)
plot(x)
```

pmvnorm

*Multivariate Normal Distribution***Description**

Computes the distribution function of the multivariate normal distribution for arbitrary limits and correlation matrices based on algorithms by Genz and Bretz.

**Usage**

```
pmvnorm(lower=-Inf, upper=Inf, mean=rep(0, length(lower)),
        corr=NULL, sigma=NULL, algorithm = GenzBretz(), ...)
```

**Arguments**

lower	the vector of lower limits of length n.
upper	the vector of upper limits of length n.
mean	the mean vector of length n.
corr	the correlation matrix of dimension n.
sigma	the covariance matrix of dimension n. Either <code>corr</code> or <code>sigma</code> can be specified. If <code>sigma</code> is given, the problem is standardized. If neither <code>corr</code> nor <code>sigma</code> is given, the identity matrix is used for <code>sigma</code> .
algorithm	an object of class <code>GenzBretz</code> or <code>Miwa</code> specifying both the algorithm to be used as well as the associated hyper parameters.
...	additional parameters (currently given to <code>GenzBretz</code> for backward compatibility issues).

**Details**

This program involves the computation of multivariate normal probabilities with arbitrary correlation matrices. It involves both the computation of singular and nonsingular probabilities. The methodology is described in Genz (1992, 1993).

Note that both `-Inf` and `+Inf` may be specified in `lower` and `upper`. For more details see [pmvt](#).

The multivariate normal case is treated as a special case of [pmvt](#) with `df=0` and univariate problems are passed to [pnorm](#).

The multivariate normal density and random deviates are available using [dmvnorm](#) and [rmvnorm](#).

**Value**

The evaluated distribution function is returned with attributes

error	estimated absolute error and
msg	status messages.

**Source**

<http://www.sci.wsu.edu/math/faculty/genz/homepage>

**References**

Genz, A. (1992). Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, **1**, 141–150.

Genz, A. (1993). Comparison of methods for the computation of multivariate normal probabilities. *Computing Science and Statistics*, **25**, 400–405.

Genz, A. and Bretz, F. (2009), *Computation of Multivariate Normal and t Probabilities*. Lecture Notes in Statistics, Vol. 195. Springer-Verlag, Heidelberg.

Miwa, A., Hayter J. and Kuriki, S. (2003). The evaluation of general non-centred orthant probabilities. *Journal of the Royal Statistical Society, Ser. B*, **65**, 223–234.

**See Also**

[qmvnorm](#)

**Examples**

```
n <- 5
mean <- rep(0, 5)
lower <- rep(-1, 5)
upper <- rep(3, 5)
corr <- diag(5)
corr[lower.tri(corr)] <- 0.5
corr[upper.tri(corr)] <- 0.5
prob <- pmvnorm(lower, upper, mean, corr)
print(prob)

stopifnot(pmvnorm(lower=-Inf, upper=3, mean=0, sigma=1) == pnorm(3))

a <- pmvnorm(lower=-Inf, upper=c(.3, .5), mean=c(2, 4), diag(2))

stopifnot(round(a, 16) == round(prod(pnorm(c(.3, .5), c(2, 4))), 16))

a <- pmvnorm(lower=-Inf, upper=c(.3, .5, 1), mean=c(2, 4, 1), diag(3))

stopifnot(round(a, 16) == round(prod(pnorm(c(.3, .5, 1), c(2, 4, 1))), 16))

# Example from R News paper (original by Genz, 1992):

m <- 3
sigma <- diag(3)
sigma[2,1] <- 3/5
sigma[3,1] <- 1/3
sigma[3,2] <- 11/15
pmvnorm(lower=rep(-Inf, m), upper=c(1, 4, 2), mean=rep(0, m), corr=sigma)
```

```
# Correlation and Covariance

a <- pmvnorm(lower=-Inf, upper=c(2,2), sigma = diag(2)*2)
b <- pmvnorm(lower=-Inf, upper=c(2,2)/sqrt(2), corr=diag(2))
stopifnot(all.equal(round(a,5) , round(b, 5)))
```

pmt

*Multivariate t Distribution***Description**

Computes the the distribution function of the multivariate t distribution for arbitrary limits, degrees of freedom and correlation matrices based on algorithms by Genz and Bretz.

**Usage**

```
pmt(lower=-Inf, upper=Inf, delta=rep(0, length(lower)),
     df=1, corr=NULL, sigma=NULL, algorithm = GenzBretz(), ...)
```

**Arguments**

lower	the vector of lower limits of length n.
upper	the vector of upper limits of length n.
delta	the vector of noncentrality parameters of length n.
df	degree of freedom as integer.
corr	the correlation matrix of dimension n.
sigma	the covariance matrix of dimension n. Either <code>corr</code> or <code>sigma</code> can be specified. If <code>sigma</code> is given, the problem is standardized. If neither <code>corr</code> nor <code>sigma</code> is given, the identity matrix is used for <code>sigma</code> .
algorithm	an object of class <code>GenzBretz</code> defining the hyper parameters of this algorithm.
...	

**Details**

This program involves the computation of central and noncentral multivariate t-probabilities with arbitrary correlation matrices. It involves both the computation of singular and nonsingular probabilities. The methodology is described in Genz and Bretz (1999, 2002).

For a given correlation matrix `corr`, for short  $A$ , say, (which has to be positive semi-definite) and degrees of freedom  $\nu$  the following values are numerically evaluated

$$I = 2^{1-\nu/2} / \Gamma(\nu/2) \int_0^\infty s^{\nu-1} \exp(-s^2/2) \Phi(s \cdot lower / \sqrt{\nu} - \delta, s \cdot upper / \sqrt{\nu} - \delta) ds$$

where

$$\Phi(a, b) = (\det(A)(2\pi)^m)^{-1/2} \int_a^b \exp(-x'Ax/2) dx$$

is the multivariate normal distribution and  $m$  is the number of rows of  $A$ .

Note that both `-Inf` and `+Inf` may be specified in the lower and upper integral limits in order to compute one-sided probabilities. Randomized quasi-Monte Carlo methods are used for the computations.

Univariate problems are passed to `pt`. If `df = 0`, normal probabilities are returned.

### Value

The evaluated distribution function is returned with attributes

<code>error</code>	estimated absolute error and
<code>msg</code>	status messages.

### Source

<http://www.sci.wsu.edu/math/faculty/genz/homepage>

### References

Genz, A. and Bretz, F. (1999), Numerical computation of multivariate t-probabilities with application to power calculation of multiple contrasts. *Journal of Statistical Computation and Simulation*, **63**, 361–378.

Genz, A. and Bretz, F. (2002), Methods for the computation of multivariate t-probabilities. *Journal of Computational and Graphical Statistics*, **11**, 950–971.

Genz, A. and Bretz, F. (2009), *Computation of Multivariate Normal and t Probabilities*. Lecture Notes in Statistics, Vol. 195. Springer-Verlag, Heidelberg.

Edwards D. and Berry, Jack J. (1987), The efficiency of simulation-based multiple comparisons. *Biometrics*, **43**, 913–928.

### See Also

[qmvt](#)

### Examples

```
n <- 5
lower <- -1
upper <- 3
df <- 4
corr <- diag(5)
corr[lower.tri(corr)] <- 0.5
delta <- rep(0, 5)
prob <- pmvt(lower=lower, upper=upper, delta=delta, df=df, corr=corr)
```

```

print(prob)

pmvt(lower=-Inf, upper=3, df = 3, sigma = 1) == pt(3, 3)

# Example from R News paper (original by Edwards and Berry, 1987)

n <- c(26, 24, 20, 33, 32)
V <- diag(1/n)
df <- 130
C <- c(1,1,1,0,0,-1,0,0,1,0,0,-1,0,0,1,0,0,0,-1,-1,0,0,-1,0,0)
C <- matrix(C, ncol=5)
### covariance matrix
cv <- C %*% V %*% t(C)
### correlation matrix
dv <- t(1/sqrt(diag(cv)))
cr <- cv * (t(dv) %*% dv)
delta <- rep(0,5)

myfct <- function(q, alpha) {
  lower <- rep(-q, ncol(cv))
  upper <- rep(q, ncol(cv))
  pmvt(lower=lower, upper=upper, delta=delta, df=df,
        corr=cr, abseps=0.0001) - alpha
}

round(uniroot(myfct, lower=1, upper=5, alpha=0.95)$root, 3)

# compare pmvt and pmvnorm for large df:

a <- pmvnorm(lower=-Inf, upper=1, mean=rep(0, 5), corr=diag(5))
b <- pmvt(lower=-Inf, upper=1, delta=rep(0, 5), df=rep(300,5),
          corr=diag(5))

a
b

stopifnot(round(a, 2) == round(b, 2))

# correlation and covariance matrix

a <- pmvt(lower=-Inf, upper=2, delta=rep(0,5), df=3,
          sigma = diag(5)*2)
b <- pmvt(lower=-Inf, upper=2/sqrt(2), delta=rep(0,5),
          df=3, corr=diag(5))
attributes(a) <- NULL
attributes(b) <- NULL
a
b
stopifnot(all.equal(round(a,3) , round(b, 3)))

a <- pmvt(0, 1,df=10)
attributes(a) <- NULL
b <- pt(1, df=10) - pt(0, df=10)
stopifnot(all.equal(round(a,10) , round(b, 10)))

```

qmvnorm

*Quantiles of the Multivariate Normal Distribution***Description**

Computes the equicoordinate quantile function of the multivariate normal distribution for arbitrary correlation matrices based on an inversion of the algorithms by Genz and Bretz.

**Usage**

```
qmvnorm(p, interval = c(-10, 10), tail = c("lower.tail",
      "upper.tail", "both.tails"), mean = 0, corr = NULL,
      sigma = NULL, algorithm = GenzBretz(), ...)
```

**Arguments**

p	probability.
interval	a vector containing the end-points of the interval to be searched by <a href="#">uniroot</a> .
tail	specifies which quantiles should be computed. <code>lower.tail</code> gives the quantile $x$ for which $P[X \leq x] = p$ , <code>upper.tail</code> gives $x$ with $P[X > x] = p$ and <code>both.tails</code> leads to $x$ with $P[-x \leq X \leq x] = p$ .
mean	the mean vector of length $n$ .
corr	the correlation matrix of dimension $n$ .
sigma	the covariance matrix of dimension $n$ . Either <code>corr</code> or <code>sigma</code> can be specified. If <code>sigma</code> is given, the problem is standardized. If neither <code>corr</code> nor <code>sigma</code> is given, the identity matrix is used for <code>sigma</code> .
algorithm	an object of class <a href="#">GenzBretz</a> or <a href="#">Miwa</a> specifying both the algorithm to be used as well as the associated hyper parameters.
...	additional parameters to be passed to <a href="#">uniroot</a> .

**Details**

Only equicoordinate quantiles are computed, i.e., the quantiles in each dimension coincide. Currently, the distribution function is inverted by using the [uniroot](#) function which may result in limited accuracy of the quantiles.

**Value**

A list with four components: `quantile` and `f.quantile` give the location of the quantile and the value of the function evaluated at that point. `iter` and `estim.prec` give the number of iterations used and an approximate estimated precision from [uniroot](#).

**See Also**

[pmvnorm](#), [qmv](#)

**Examples**

```
qmvnorm(0.95, sigma = diag(2), tail = "both")
```

---

qmv
*Quantiles of the Multivariate t Distribution*


---

**Description**

Computes the equicoordinate quantile function of the multivariate t distribution for arbitrary correlation matrices based on an inversion of the algorithms by Genz and Bretz.

**Usage**

```
qmv(p, interval = c(-10, 10), tail = c("lower.tail",
  "upper.tail", "both.tails"), df = 1, delta = 0, corr = NULL,
  sigma = NULL, algorithm = GenzBretz(), ...)
```

**Arguments**

p	probability.
interval	a vector containing the end-points of the interval to be searched by <a href="#">uniroot</a> .
tail	specifies which quantiles should be computed. <code>lower.tail</code> gives the quantile $x$ for which $P[X \leq x] = p$ , <code>upper.tail</code> gives $x$ with $P[X > x] = p$ and <code>both.tails</code> leads to $x$ with $P[-x \leq X \leq x] = p$ .
delta	the vector of noncentrality parameters of length $n$ .
df	degree of freedom as integer.
corr	the correlation matrix of dimension $n$ .
sigma	the covariance matrix of dimension $n$ . Either <code>corr</code> or <code>sigma</code> can be specified. If <code>sigma</code> is given, the problem is standardized. If neither <code>corr</code> nor <code>sigma</code> is given, the identity matrix is used for <code>sigma</code> .
algorithm	an object of class <a href="#">GenzBretz</a> defining the hyper parameters of this algorithm.
...	additional parameters to be passed to <a href="#">uniroot</a> .

**Details**

Only equicoordinate quantiles are computed, i.e., the quantiles in each dimension coincide. Currently, the distribution function is inverted by using the [uniroot](#) function which may result in limited accuracy of the quantiles.

**Value**

A list with four components: `quantile` and `f.quantile` give the location of the quantile and the value of the function evaluated at that point. `iter` and `estim.prec` give the number of iterations used and an approximate estimated precision from `uniroot`.

**See Also**

[pmvnorm](#), [qmvf](#)

**Examples**

```
qmvf(0.95, df = 16, tail = "both")
```

# Index

## \*Topic **distribution**

- algorithms, 1
- Mvnorm, 2
- Mvt, 3
- pmvnorm, 4
- pmvt, 6
- qmvnorm, 9
- qmvt, 10

## \*Topic **multivariate**

- Mvnorm, 2
- Mvt, 3

algorithms, 1

dmvnorm, 5  
dmvnorm (*Mvnorm*), 2  
dmvt (*Mvt*), 3

GenzBretz, 4, 7, 9, 11  
GenzBretz (*algorithms*), 1

Miwa, 4, 9  
Miwa (*algorithms*), 1  
Mvnorm, 2  
Mvt, 3

pmvnorm, 3, 4, 10, 11  
pmvt, 4, 5, 6  
pnorm, 5  
pt, 7

qmvnorm, 3, 5, 9  
qmvt, 4, 8, 10, 10, 11

rmvnorm, 5  
rmvnorm (*Mvnorm*), 2  
rmvt (*Mvt*), 3  
rnorm, 3

uniroot, 9–11