

# Package ‘nntmvn’

July 22, 2025

**Title** Draw Samples of Truncated Multivariate Normal Distributions

**Version** 1.2.0

**Date** 2025-02-06

**Author** Jian Cao [aut, cre],  
Matthias Katzfuss [aut]

**Maintainer** Jian Cao <jcao2416@gmail.com>

**License** GPL (>= 2)

**Description** Draw samples from truncated multivariate normal distribution using the sequential nearest neighbor (SNN) method introduced in ``Scalable Sampling of Truncated Multivariate Normals Using Sequential Nearest-Neighbor Approximation" <[doi:10.48550/arXiv.2406.17307](https://doi.org/10.48550/arXiv.2406.17307)>.

**Encoding** UTF-8

**Imports** GpGp, TruncatedNormal, RANN

**LinkingTo** Rcpp

**RoxygenNote** 7.3.2

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2025-02-08 18:50:02 UTC

## Contents

|                   |          |
|-------------------|----------|
| corr_nn . . . . . | 2        |
| nntmvn . . . . .  | 2        |
| rptmvn . . . . .  | 3        |
| rtmvn . . . . .   | 4        |
| <b>Index</b>      | <b>7</b> |

---

|                      |                                                                                                                                                                                                                                                                                       |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>corr_nn</code> | <i>Find ordered nearest neighbors based on correlation, assuming the absolute value of the correlation is monotonically decreasing with distance. Returns an <math>n \times (m + 1)</math> matrix, each row indicating the <math>m + 1</math> nearest neighbors including itself.</i> |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

---

### Description

Find ordered nearest neighbors based on correlation, assuming the absolute value of the correlation is monotonically decreasing with distance. Returns an  $n \times (m + 1)$  matrix, each row indicating the  $m + 1$  nearest neighbors including itself.

### Usage

```
corr_nn(covmat, m)
```

### Arguments

|                     |                                 |
|---------------------|---------------------------------|
| <code>covmat</code> | the covariance matrix           |
| <code>m</code>      | the number of nearest neighbors |

### Value

an  $n \times (m + 1)$  matrix

### Examples

```
library(RANN)
library(nntmvn)
set.seed(123)
d <- 3
n <- 100
locs <- matrix(runif(d * n), n, d)
covparms <- c(2, 0.01, 0)
covmat <- GpGp::matern15_isotropic(covparms, locs)
m <- 10
NNarray_test <- RANN::nn2(locs, k = m + 1)[[1]]
NNarray <- nntmvn::corr_nn(covmat, m)
cat("Number of mismatch is", sum(NNarray != NNarray_test, na.rm = TRUE))
```

---

nntmvn

*nntmvn*

---

### Description

Draw Samples of Truncated Multivariate Normal Distributions

---

|        |                                                                                                                                                |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------|
| rptmvn | <i>Draw one sample of the underlying GP responses for a partially censored Gaussian process using sequential nearest neighbor (SNN) method</i> |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------|

---

## Description

Draw one sample of the underlying GP responses for a partially censored Gaussian process using sequential nearest neighbor (SNN) method

## Usage

```
rptmvn(
  y,
  cens_lb,
  cens_ub,
  mask_cens,
  m = 30,
  covmat = NULL,
  locs = NULL,
  cov_name = NULL,
  cov_parm = NULL,
  NN = NULL,
  ordering = 0,
  seed = NULL
)
```

## Arguments

|           |                                                                                                           |
|-----------|-----------------------------------------------------------------------------------------------------------|
| y         | uncensored responses of length n, where n is the number of all responses                                  |
| cens_lb   | lower bound vector for TMVN of length n                                                                   |
| cens_ub   | upper bound vector for TMVN of length n                                                                   |
| mask_cens | mask for censored responses (also locations) of length n                                                  |
| m         | positive integer for the number of nearest neighbors used                                                 |
| covmat    | n-by-n dense covariance matrix, either covmat or locs, cov_name, and cov_parms need to be provided        |
| locs      | location matrix n X d                                                                                     |
| cov_name  | covariance function name from the GpGp package                                                            |
| cov_parm  | parameters for the covariance function from the GpGp package                                              |
| NN        | n X m matrix for nearest neighbors. i-th row is the nearest neighbor indices of y_i. NN[i, 1] should be i |
| ordering  | 0 for do not reorder, 1 for variance descending order, 2 for maximin ordering                             |
| seed      | set seed for reproducibility                                                                              |

**Value**

a vector of length  $n$  representing the underlying GP responses

**Examples**

```
library(GpGp)
library(RANN)
library(nntmvn)
set.seed(123)
x <- matrix(seq(from = 0, to = 1, length.out = 51), ncol = 1)
cov_name <- "matern15_isotropic"
cov_parm <- c(1.0, 0.1, 0.001) #' variance, range, nugget
cov_func <- getFromNamespace(cov_name, "GpGp")
covmat <- cov_func(cov_parm, x)
y <- t(chol(covmat)) %*% rnorm(length(x))
mask <- y < 0.3
y_cens <- y
y_cens[mask] <- NA
lb <- rep(-Inf, 100)
ub <- rep(0.3, 100)
m <- 10
y_samp_mtd1 <- rptmvn(y_cens, lb, ub, mask,
  m = m, locs = x,
  cov_name = cov_name, cov_parm = cov_parm, seed = 123
)
y_samp_mtd2 <- rptmvn(y_cens, lb, ub, mask,
  m = m, covmat = covmat,
  seed = 123
)
plot(x, y_cens, ylim = range(y))
points(x[mask, ], y[mask], col = "blue")
plot(x, y_cens, ylim = range(y))
points(x[mask, ], y_samp_mtd1[mask], col = "red")
plot(x, y_cens, ylim = range(y))
points(x[mask, ], y_samp_mtd2[mask], col = "brown")
```

---

rtmvn

---

*Draw one sample from a truncated multivariate normal (TMVN) distribution using sequential nearest neighbor (SNN) method*


---

**Description**

Draw one sample from a truncated multivariate normal (TMVN) distribution using sequential nearest neighbor (SNN) method

**Usage**

```
rtmvn(
  cens_lb,
  cens_ub,
  m = 30,
  covmat = NULL,
  locs = NULL,
  cov_name = NULL,
  cov_parm = NULL,
  NN = NULL,
  ordering = 0,
  seed = NULL
)
```

**Arguments**

|          |                                                                                                           |
|----------|-----------------------------------------------------------------------------------------------------------|
| cens_lb  | lower bound vector for TMVN of length n                                                                   |
| cens_ub  | upper bound vector for TMVN of length n                                                                   |
| m        | positive integer for the number of nearest neighbors used                                                 |
| covmat   | n-by-n dense covariance matrix, either covmat or locs, cov_name, and cov_parms need to be provided        |
| locs     | location matrix n X d                                                                                     |
| cov_name | covariance function name from the GpGp package                                                            |
| cov_parm | parameters for the covariance function from the GpGp package                                              |
| NN       | n X m matrix for nearest neighbors. i-th row is the nearest neighbor indices of y_i. NN[i, 1] should be i |
| ordering | 0 for do not reorder, 1 for variance descending order, 2 for maximin ordering                             |
| seed     | set seed for reproducibility                                                                              |

**Value**

a vector of length n representing the underlying GP responses

**Examples**

```
library(nntmvn)
library(TruncatedNormal)
set.seed(123)
x <- matrix(seq(from = 0, to = 1, length.out = 51), ncol = 1)
cov_name <- "matern15_isotropic"
cov_parm <- c(1.0, 0.1, 0.001) #' variance, range, nugget
cov_func <- getFromNamespace(cov_name, "GpGp")
covmat <- cov_func(cov_parm, x)
lb <- rep(-Inf, nrow(x))
ub <- rep(-1, nrow(x))
m <- 30
samp_SNN <- matrix(NA, 3, nrow(x))
```

```
for (i in 1:3) {  
  samp_SNN[i, ] <- nntmvn::rtmvn(lb, ub, m = m, covmat = covmat, locs = x, ordering = 0)  
}  
samp_TN <- TruncatedNormal::rtmvnorm(3, rep(0, nrow(x)), covmat, lb, ub)  
qqplot(samp_SNN, samp_TN, xlim = range(samp_SNN, samp_TN), ylim = range(samp_SNN, samp_TN))  
abline(a = 0, b = 1, lty = "dashed", col = "red")
```

# Index

`corr_nn`, [2](#)

`nntmvn`, [2](#)

`rptmvn`, [3](#)

`rtmvn`, [4](#)