

Package ‘NormGen’

October 22, 2026

Title An Efficient and Precise Quantile Mapping for Normal
Pseudo-Random Number Generation

Version 0.0.6

Description High-performance pseudo-normal random generator using an optimized quantile mapping algorithm. It provides fast generation of normal pseudo-random numbers based on precise mathematical approximations. The package features `Rnorm()` for efficient single-threaded generation and `parallel_Rnorm()`, a high-performance, multithreaded variant powered by 'RcppParallel' and Intel 'TBB'. Package compilation flags are optimized to support multi-core parallel processing.

Depends R ($\geq 3.5.0$)

Imports Rcpp ($\geq 1.0.0$), RcppParallel

LinkingTo Rcpp, RcppParallel

Suggests testthat ($\geq 3.0.0$), knitr, rmarkdown

VignetteBuilder knitr

License GPL-3

Language en-US

Encoding UTF-8

Config/roxygen2/version 8.1.0

NeedsCompilation yes

Author Piotr Sulewski [aut, cre]

Maintainer Piotr Sulewski <piotr.sulewski@ups1.edu.pl>

Repository CRAN

Date/Publication 2026-10-22 16:20:08 UTC

Contents

parallel_Rnorm	2
Rnorm	3
Index	4

parallel_Rnorm	<i>High-Performance Multithreaded Pseudo-Normal Random Number Generator</i>
----------------	---

Description

Generates multi-core parallel processing normal pseudo-random numbers powered by 'RcppParallel' and Intel 'TBB'.

Usage

```
parallel_Rnorm(n, mean = 0, sd = 1, seed = 12345L)
```

Arguments

n	An integer specifying the number of observations to generate.
mean	A numeric value specifying the mean of the distribution. Default is 0.
sd	A numeric value specifying the standard deviation of the distribution. Default is 1.
seed	An integer to initialize the pseudo-random number generator state. Default is 12345L.

Details

This function divides the generation task across available CPU threads to maximize execution performance. By passing a specific integer to the seed parameter, you ensure the full reproducibility of the obtained results across different executions.

Value

A numeric vector containing n pseudo-random numbers drawn from the specified normal distribution.

Author(s)

Piotr Sulewski <piotr.sulewski@upsl.edu.pl>

Examples

```
# Generate parallel numbers with an explicit seed
parallel_samples <- parallel_Rnorm(1000, mean = 0, sd = 1, seed = 2026)

# Verify reproducibility directly via the parameter
verify_samples <- parallel_Rnorm(1000, mean = 0, sd = 1, seed = 2026)
identical(parallel_samples, verify_samples)
```

Rnorm*Single-threaded Pseudo-Normal Random Number Generator*

Description

Generates high-performance normal pseudo-random numbers based on an optimized quantile mapping algorithm.

Usage

```
Rnorm(n, m = 0, s = 1)
```

Arguments

n	An integer specifying the number of observations to generate.
m	A numeric value specifying the mean of the distribution. Default is 0.
s	A numeric value specifying the standard deviation of the distribution. Default is 1.

Details

This function applies precise mathematical approximations to provide a fast alternative to the classic `rnorm` function.

Value

A numeric vector containing `n` pseudo-random numbers drawn from the specified normal distribution.

Author(s)

Piotr Sulewski <piotr.sulewski@upsl.edu.pl>

References

Sulewski P., Drapella A. (2026). *An Efficient and Precise Quantile Mapping for Normal Pseudo-Random Number Generation*. Journal of Statistical Software, Under review.

Examples

```
# Generate 100 random numbers from N(0,1)
samples <- Rnorm(100)
mean(samples)
sd(samples)

# Generate with custom mean and standard deviation
custom_samples <- Rnorm(10, m = 2, s = 2)
```

Index

`parallel_Rnorm`, [2](#)

`Rnorm`, [3](#)