# Package 'rrandvec'

July 23, 2025

**Title** Generate Random Vectors Whose Components Sum Up to One

**Description** A single method implementing multiple approaches to generate pseudo-random vectors whose components sum up to one (see, e.g., Maziero (2015) <doi:10.1007/s13538-015-0337-8>). The components of such vectors can for example be used for weighting objectives when reducing multi-objective optimisation problems to a single-objective problem in the socalled weighted sum scalarisation approach.

**Version** 1.0.0

**Depends** R (>= 3.1.0)

**Imports** Rcpp, checkmate

**Suggests** covr, testthat, scatterplot3d

**License** BSD_2_clause + file LICENSE

**URL** https://jakobbossek.github.io/rrandvec/, https://github.com/jakobbossek/rrandvec

**BugReports** https://github.com/jakobbossek/rrandvec/issues

**Encoding** UTF-8

**ByteCompile** true

**RoxygenNote** 7.2.3

**LinkingTo** Rcpp

**NeedsCompilation** yes

**Author** Jakob Bossek [aut, cre, cph] (ORCID: <https://orcid.org/0000-0002-4121-4668>)

**Maintainer** Jakob Bossek <j.bossek@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-03-30 07:10:02 UTC

## Contents

rrandvec *Generate random vectors that sum up to one.*

## Description

Generate an $n \times d$ matrix. Each row vector is a probability vector $(p_1, \ldots, p_d)$ with $\sum_{i=1}^{d} p_i = 1$. The function offers several methods to generate the rows in a way that the components are unbiased which means that they are required to have similar / the same probability distributions.

[1] Maziero, J. Generating Pseudo-Random Discrete Probability Distributions. Brazilian Journal of Physics 45, 377–382 (2015). https://doi.org/10.1007/s13538-015-0337-8

[2] Grimme, C. Picking a Uniformly Random Point from an Arbitrary Simplex. Technical Report. https://doi.org/10.13140/RG.2.1.3807.6968

## Usage

```
rrandvec(n, d, method = "normalization", shuffle = FALSE, as.df = FALSE)
```

## Arguments

| | |
|---|---|
| n | [integer(1)]<br>Number of vectors to generate. |
| d | [integer(1)]<br>Number of components of each vector (at least 2). |
| method | [character(1)]<br>One of "norm" (normalization method), "trigonometric", "simplex" (sample from a unit simplex), "exponential" or "iterative". Default is simplex. |
| shuffle | [logical(1)]<br>Should the values of each vector be permutatet randomly? Background: methods "iterative" and "trigonometric" introduce unwanted bias (see desciption). This issue can be alliviated by random shuffling. Default is FALSE. |
| as.df | [logical(1)]<br>Should the return value be a data frame with column names X1 to Xd? Default is FALSE. |

## Value

[matrix(n, d)] $(n \times d)$ matrix even if $n = 1$.

## Examples

```
R = rrandvec(1000, 2)
R = rrandvec(1000, 5, method ="iterative")
R = rrandvec(1000, 3, method = "trigonometric", shuffle = TRUE, as.df = TRUE)

if (require("scatterplot3d")) {
  scatterplot3d::scatterplot3d(R, angle = 120, cex.symbols = 0.5, pch = 3, color = "blue")
}
```

# Index