

Package ‘simdd’

July 23, 2025

Type Package

Title Simulation of Fisher Bingham and Related Directional Distributions

Version 1.1-2

Date 2023-10-31

Description Simulation methods for the Fisher Bingham distribution on the unit sphere, the matrix Bingham distribution on a Grassmann manifold, the matrix Fisher distribution on $SO(3)$, and the bivariate von Mises sine model on the torus. The methods use an acceptance/rejection simulation algorithm for the Bingham distribution and are described fully by Kent, Ganeiber and Mardia (2018) <[doi:10.1080/10618600.2017.1390468](https://doi.org/10.1080/10618600.2017.1390468)>. These methods supersede earlier MCMC simulation methods and are more general than earlier simulation methods. The methods can be slower in specific situations where there are existing non-MCMC simulation methods (see Section 8 of Kent, Ganeiber and Mardia (2018) <[doi:10.1080/10618600.2017.1390468](https://doi.org/10.1080/10618600.2017.1390468)> for further details).

License GPL-2

Suggests CircStats, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author John Kent [aut, cph],
Kassel Liam Hingee [cre]

Maintainer Kassel Liam Hingee <kassel.hingee@anu.edu.au>

Repository CRAN

Date/Publication 2023-10-31 03:10:03 UTC

Contents

simdd-package	2
Simulate Fisher Bingham and related distributions	2

Index	6
--------------	----------

 simdd-package

simdd – Simulation of Fisher Bingham and Related Directional Distributions

Description

Simulation methods for the Fisher Bingham distribution on the unit sphere, the matrix Bingham distribution on a Grassmann manifold, the matrix Fisher distribution on $SO(3)$, and the bivariate von Mises sine model on the torus. The methods use the first ever general purpose acceptance/rejection simulation algorithm for the Bingham distribution and are described fully by Kent, Ganeiber and Mardia (2018). These methods superseded earlier MCMC simulation methods and are more general than earlier simulation methods. The methods can be slower in specific situations where there are existing non-MCMC simulation methods (see Section 8 of Kent, Ganeiber and Mardia (2018) for further details).

Author(s)

John Kent <j.t.kent@leeds.ac.uk>

References

John T. Kent, Asaad M. Ganeiber & Kanti V. Mardia (2018) A New Unified Approach for the Simulation of a Wide Class of Directional Distributions, *Journal of Computational and Graphical Statistics*, 27:2, 291-301, DOI: [doi:10.1080/10618600.2017.1390468](https://doi.org/10.1080/10618600.2017.1390468)

Examples

```
# simulate 20 random unit vectors from a Bingham distribution
# with concentration matrix diag(c(1,2,-3)).
X=rBingham(20,c(1,2,-3))
# check the efficiency of the simulation algorithm
attr(X,"summary")
```

 Simulate Fisher Bingham and related distributions

Simulate Fisher Bingham and related distributions

Description

Simulate the Fisher Bingham distribution on the unit sphere, the matrix Bingham distribution on a Grassmann manifold, the matrix Fisher distribution on $SO(3)$, and the bivariate von Mises sine model on the torus = product of two circles.

Usage

```

rFisherBingham(nsim, mu = 0, Aplus = 0, q = dimset(mu, Aplus), mtop = 1000)
rBingham(nsim, Aplus, q = dimq(Aplus), mtop = 1000)
rBingham.Grassmann(nsim, Aplus = 0, q = dimq(Aplus), r = 1, mtop = 1000)
rFisher.S03(nsim, Fmat, mtop = 1000)
rvMsin.torus(nsim, k1, k2, alpha, mtop=1000)
rBessel(nsim, k1, k2, alpha, mtop=1000)

```

Arguments

nsim	the desired number of simulations
mu	a vector of length q . If μ is entered as a scalar, it is interpreted as the vector $(0, \dots, 0, \mu)$ with zeros except in the final coordinate.
Aplus	a symmetric q by q matrix. If Aplus is given as a scalar, it is interpreted as the q by q matrix of zeros. If Aplus is given as a vector, it is interpreted as the matrix $\text{diag}(\text{Aplus})$.
q	The simulation is done on the unit sphere in R^q . Thus for the circle, $q=2$. Note that q only needs to be given explicitly if both μ and Aplus are scalars. In the default setting the internal function dimset , $q=\text{dimset}(\mu, \text{Aplus})$, determines q from the length of μ or the dimension of Aplus, if it can.
r	For <code>rBingham.Grassmann</code> , r denotes the number of columns ($1 \leq r \leq q-1$) in the matrix Bingham distribution
mtop	The maximum number of attempts to generate nsim simulations. A finite mtop prevents infinite loops in extreme situations. If mtop is reached there will be a warning.
Fmat	The 3 by 3 parameter matrix for the matrix Fisher distribution in the function <code>rFisher.S03</code> .
k1, k2, alpha	The two concentration parameters and the interaction parameter in the function <code>rvMsin.torus</code> .

Details

The Fisher Bingham distribution on the unit sphere in R^q has density proportional to

$$\exp(\mu^T x + x^T \text{Aplus} x)$$

where x is a unit vector in R^q , and μ (q -dimensional vector) and Aplus (q by q symmetric) are parameters.

The matrix Bingham distribution on q by r matrices X whose columns are orthonormal, is given by the density proportional to

$$\exp(\text{trace}(X^T \text{Aplus} X)).$$

The Bingham distribution on the unit sphere in R^q can be simulated using (a) `rBingham`, (b) `rFisherBingham` with $\mu=0$, and (c) `rBingham.Grassmann` with $r=1$. Choice (a) will be fastest.

The Fisher distribution can be simulated using `rFisherBingham` with `Aplus=0`.

The matrix Fisher distribution on $SO(3)$ has density proportional to

$$\exp(\text{trace}(\text{Fmat}^T X))$$

where X is a 3 by 3 rotation matrix, and Fmat is a 3 by 3 parameter matrix.

The bivariate von Mises sine model on the torus has density proportional to

$$\exp(k_1 \cos(\theta) + k_2 \cos(\phi) + \alpha \sin(\theta) \sin(\phi))$$

for two angles θ, ϕ . The Bessel density is obtained from the bivariate von Mises sine model as the marginal density of θ .

If `mtop` is reached before obtaining `nsim` simulations then a warning is created and the returned array will have fewer than `nsim` rows.

Value

For `rBingham` and `rFisherBingham`, the output is an `nsim` by `q` matrix. Each row is a simulated unit vector.

For `rBingham.Grassmann`, the output is an `nsim` by `q` by `r` array. For each value of the first index, the result is a simulated `q` by `r` matrix with orthonormal columns.

For `rFisher.S03`, the output is an `nsim` by `r` by `r` array. For each value of the first index, the result is a simulated `r` by `r` matrix rotation matrix.

For `rvMsin.torus`, the output is an `nsim` by 4 matrix, with each row containing the simulated value of $(\cos(\theta), \sin(\theta), \cos(\phi), \sin(\phi))$.

For `rBessel`, the output is an `nsim` by 2 matrix containing the marginal simulated values of $(\cos(\theta), \sin(\theta))$.

In all cases, the output has an attribute `summary`, which is a vector of length 6 summarizing some details about the number of simulations needed in the acceptance/rejection algorithm. The key element of this vector is called `efficiency`, a number between 0 and 1, where 1 means that all the simulated values from the envelope distribution have been accepted.

- `ntry` is the number of simulations drawn from the envelope distribution.
- `efficiency` is the proportion of simulations drawn from the envelope distribution that were accepted.
- `success` is 1 when simulations were completed, and 0 otherwise. Usually the simulations are incomplete because the number of iterations (in entry `mloops`) has reached the maximum `mtop`.
- `mloops` is the number of iterations used.
- `minfg` is the smallest observed value of the envelope.
- `maxfg` is the largest observed value of the envelope.

Author(s)

J T Kent

References

John T. Kent, Asaad M. Ganeiber & Kanti V. Mardia (2018) A New Unified Approach for the Simulation of a Wide Class of Directional Distributions, *Journal of Computational and Graphical Statistics*, 27:2, 291-301, DOI: [doi:10.1080/10618600.2017.1390468](https://doi.org/10.1080/10618600.2017.1390468)

See Also

The function `rvm` in the CRAN library `CircStats` provides a more efficient method to simulate from the von Mises distribution (i.e. the Fisher distribution with $q=2$) than `rFisherBingham`.

Examples

```
X1=rBingham(10,c(1,2,-3))
attr(X1,"summary")
X2=rFisherBingham(10,c(3,2,1),c(1,2,-3))
X3=rBingham.Grassmann(10,c(1,2,-3),r=2)
X4=rFisher.S03(10,matrix(c(1,2,3,4,2,1,0,1,2),3,3))
X5=rvmSin.torus(10,2.2,3.1,1.3)
X6=rBessel(10,2.2,3.1,1.3)
```

Index

rBessel (Simulate Fisher Bingham and related distributions), [2](#)
rBingham (Simulate Fisher Bingham and related distributions), [2](#)
rFisher.S03 (Simulate Fisher Bingham and related distributions), [2](#)
rFisherBingham (Simulate Fisher Bingham and related distributions), [2](#)
rvm, [5](#)
rvMsin.torus (Simulate Fisher Bingham and related distributions), [2](#)

simdd (simdd-package), [2](#)
simdd-package, [2](#)
Simulate Fisher Bingham and related distributions, [2](#)