

Package ‘simpleMH’

July 23, 2025

Title Simple Metropolis-Hastings MCMC Algorithm

Version 0.1.1

Description A very bare-bones interface to use the Metropolis-Hastings Monte Carlo Markov Chain algorithm. It is suitable for teaching and testing purposes.

Imports mvtnorm

Suggests coda, mockery, testthat (>= 3.0.0), knitr, rmarkdown

License GPL-3

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

RoxygenNote 7.1.1.9001

VignetteBuilder knitr

URL <https://github.com/Bisaloo/simpleMH>

BugReports <https://github.com/Bisaloo/simpleMH/issues>

NeedsCompilation no

Author Hugo Gruson [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0002-4094-1476>>)

Maintainer Hugo Gruson <hugo.gruson+R@normalesup.org>

Repository CRAN

Date/Publication 2024-02-06 18:20:02 UTC

Contents

simpleMH	2
Index	3

simpleMH

*Simple Metropolis-Hastings MCMC***Description**

Simple Metropolis-Hastings MCMC

Usage

```
simpleMH(f, inits, theta.cov, max.iter, coda = FALSE, ...)
```

Arguments

<code>f</code>	function that returns a single scalar value proportional to the log probability density to sample from.
<code>inits</code>	numeric vector with the initial values for the parameters to estimate
<code>theta.cov</code>	covariance matrix of the parameters to estimate.
<code>max.iter</code>	maximum number of function evaluations
<code>coda</code>	logical. Should the samples be returned as <code>coda::mcmc</code> object? (defaults to FALSE)
<code>...</code>	further arguments passed to <code>f</code>

Value

- if `coda = FALSE` a list with:
 - *samples*: A two dimensional array of samples with dimensions generation x parameter
 - *log.p*: A numeric vector with the log density evaluate at each generation.
- if `coda = TRUE` a list with:
 - *samples*: A object of class `coda::mcmc` containing all samples.
 - *log.p*: A numeric vector with the log density evaluate at each generation.

Examples

```
p.log <- function(x) {
  B <- 0.03
  return(-x[1]^2/200 - 1/2*(x[2]+B*x[1]^2-100*B)^2)
}

simpleMH(p.log, inits=c(0, 0), theta.cov = diag(2), max.iter=3000)
```

Index

`coda::mcmc`, [2](#)

`simpleMH`, [2](#)