Package 'mcmcse'

July 4, 2017

Version 1.3-2

Date 2017-07-03

Title Monte Carlo Standard Errors for MCMC

Author James M. Flegal <jflegal@ucr.edu>,
John Hughes <j.hughes@ucdenver.edu>,
Dootika Vats <D.Vats@warwick.ac.uk>, and
Ning Dai <daixx224@umn.edu>

Maintainer Dootika Vats <D.Vats@warwick.ac.uk>

Imports utils, ellipse, Rcpp (>= 0.12.10)

LinkingTo Rcpp, RcppArmadillo

Suggests testthat, mAr, knitr

Description Provides tools for computing Monte Carlo standard errors (MCSE) in Markov chain Monte Carlo (MCMC) settings. MCSE computation for expectation and quantile estimators is supported as well as multivariate estimations. The package also provides functions for computing effective sample size and for plotting Monte Carlo estimates versus sample size.

License GPL (>= 2)

URL http://faculty.ucr.edu/~jflegal , http://johnhughes.org, http://www2.warwick.ac.uk/fac/sci/statistics/crism/visitors/vats

VignetteBuilder knitr

Repository CRAN

NeedsCompilation yes

Date/Publication 2017-07-04 07:20:37 UTC

R topics documented:

`mcmcse-package` *Monte Carlo Standard Errors for MCMC*

Description

Provides tools for computing Monte Carlo standard errors (MCSE) in Markov chain Monte Carlo (MCMC) settings. MCSE computation for expectation and quantile estimators is supported. The package also provides functions for computing effective sample size and for plotting Monte Carlo estimates versus sample size.

Details

Package:	mcmcse
Type:	Package
Version:	1.3-2
Date:	2017-07-03
License:	GPL (>= 2)

Author(s)

James M. Flegal <jflegal@ucr.edu>,\ John Hughes <j.hughes@ucdenver.edu> \ Dootika Vats <D.Vats@warwick.ac.uk> and\ Ning Dai <daixx224@umn.edu>

Maintainer: Dootika Vats <D.Vats@warwick.ac.uk>

References

Dai, N and Jones, G.L. (2017+) Multivariate initial sequence estimators in Markov chain Monte Carlo, *Journal of Multivariate Analysis*.

Flegal, J. M. (2012) Applicability of subsampling bootstrap methods in Markov chain Monte Carlo. In Wozniakowski, H. and Plaskota, L., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*, pages 363–372. Springer-Verlag.

Flegal, J. M. and Jones, G. L. (2010) Batch means and spectral variance estimators in Markov chain Monte Carlo. *The Annals of Statistics*, **38**, 1034–1070.

Flegal, J. M. and Jones, G. L. (2011) Implementing Markov chain Monte Carlo: Estimating with confidence. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X., editors, *Handbook of Markov Chain Monte Carlo*, pages 175–197. Chapman & Hall/CRC Press.

Flegal, J. M., Jones, G. L., and Neath, R. (2012) Markov chain Monte Carlo estimation of quantiles. *University of California, Riverside, Technical Report*.

Gong, L., and Flegal, J. M. A practical sequential stopping rule for high-dimensional Markov chain Monte Carlo. Journal of Computational and Graphical Statistics (to appear).

Jones, G. L., Haran, M., Caffo, B. S. and Neath, R. (2006) Fixed-width output analysis for Markov chain Monte Carlo. *Journal of the American Statistical Association*, **101**, 1537–1547.

Vats, D., Flegal, J. M., and, Jones, G. L Multivariate Output Analysis for Markov chain Monte Carlo, *arXiv preprint arXiv:1512.07713 (2015)*.

Vats, D., Flegal, J. M., and, Jones, G. L. (2017+) Strong Consistency of multivariate spectral variance estimators for Markov chain Monte Carlo, *Bernoulli*.

Examples

```
library(mAr)
p <- 3
n <- 1e3
omega <- 5*diag(1,p)

## Making correlation matrix var(1) model
set.seed(100)
foo <- matrix(rnorm(p^2), nrow = p)
foo <- foo %*% t(foo)
phi <- foo / (max(eigen(foo)$values) + 1)

out <- as.matrix(mAr.sim(rep(0,p), phi, omega, N = n))

mcse(out[,1], method = "bart")

mcse.bm <- mcse.multi(x = out)
mcse.tuk <- mcse.multi(x = out, method = "tukey")
```

`confRegion` *Confidence regions (ellipses) for Monte Carlo estimates*

Description

Constructs confidence regions (ellipses) from the Markov chain output for the features of interest. Function uses the ellipse package.

Usage

```
confRegion(mcse.obj, which = c(1,2), level = .95)
```

Arguments

mcse.obj	the list returned by the mcse.multi or mcse.initseq command
which	integer vector of length 2 indicating the component for which to make the confidence ellipse. Chooses the first two by default.
level	confidence level for the ellipse

Details

Returns a matrix of x and y coordinates for the ellipse. Use plot function on the matrix to plot the ellipse

Examples

```
library(mAr)
p <- 3
n <- 1e3
omega <- 5*diag(1,p)

## Making correlation matrix var(1) model
set.seed(100)
foo <- matrix(rnorm(p^2), nrow = p)
foo <- foo %*% t(foo)
phi <- foo / (max(eigen(foo)$values) + 1)

out <- as.matrix(mAr.sim(rep(0,p), phi, omega, N = n))
mcerror <- mcse.multi(out)

## Plotting the ellipse
plot(confRegion(mcerror), type = 'l')
```

---

ess *Estimate effective sample size (ESS) as described in Gong and Felgal (2015).*

Description

Estimate effective sample size (ESS) as described in Gong and Flegal (2015).

Usage

```
ess(x, g = NULL, ...)
```

Arguments

x	a matrix or data frame of Markov chain output. Number of rows is the Monte Carlo sample size.
...	arguments passed on to the mcse.mat function. For example method = "tukey" and size = "cuberoot" can be used.
g	a function that represents features of interest. g is applied to each row of x and thus g should take a vector input only. If g is NULL, g is set to be identity, which is estimation of the mean of the target density.

Details

ESS is the size of an iid sample with the same variance as the current sample. ESS is given by

$$\text{ESS} = n\frac{\lambda^2}{\sigma^2},$$

where λ^2 is the sample variance and σ^2 is an estimate of the variance in the CLT. This is by default the batch means estimator, but the default can be changed with the method argument.

Value

The function returns the estimated effective sample size.

References

Gong, L. and Flegal, J. M. (2015) A practical sequential stopping rule for high-dimensional Markov chain Monte Carlo *Journal of Computational and Graphical Statistics*.

See Also

minESS, which calculates the minimum effective samples required for the problem.

multiESS, which calculates multivariate effective sample size using a Markov chain and a function g.

---

estvssamp *Create a plot that shows how Monte Carlo estimates change with increasing sample size.*

Description

Create a plot that shows how Monte Carlo estimates change with increasing sample size.

Usage

```
estvssamp(x, g = mean, main = "Estimates vs Sample Size",
  add = FALSE, ...)
```

Arguments

x	a sample vector.
g	a function such that $E(g(x))$ is the quantity of interest. The default is g = mean.
main	an overall title for the plot. The default is "Estimates vs Sample Size".
add	logical. If TRUE, add to a current plot.
...	additional arguments to the plotting function.

Value

NULL

Examples

```
## Not run:
estvssamp(x, main = expression(E(beta)))
estvssamp(y, add = TRUE, lty = 2, col = "red")
## End(Not run)
```

---

mcse *Compute Monte Carlo standard errors for expectations.*

Description

Compute Monte Carlo standard errors for expectations.

Usage

```
mcse(x, size = "sqroot", g = NULL,
  method = c("bm", "obm", "tukey", "bartlett"),
  warn = FALSE)
```

Arguments

x	a vector of values from a Markov chain.
size	the batch size. The default value is "sqroot", which uses the square root of the sample size. "cuberoot" will cause the function to use the cube root of the sample size. A numeric value may be provided if neither "sqroot" nor "cuberoot" is satisfactory.
g	a function such that $E(g(x))$ is the quantity of interest. The default is NULL, which causes the identity function to be used.
method	the method used to compute the standard error. This is one of "bm" (batch means, the default), "obm" (overlapping batch means), "tukey" (spectral variance method with a Tukey-Hanning window), or "bartlett" (spectral variance method with a Bartlett window).
warn	a logical value indicating whether the function should issue a warning if the sample size is too small (less than 1,000).

Value

mcse returns a list with two elements:

est	an estimate of $E(g(x))$.
se	the Monte Carlo standard error.

References

Flegal, J. M. (2012) Applicability of subsampling bootstrap methods in Markov chain Monte Carlo. In Wozniakowski, H. and Plaskota, L., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010* (to appear). Springer-Verlag.

Flegal, J. M. and Jones, G. L. (2010) Batch means and spectral variance estimators in Markov chain Monte Carlo. *The Annals of Statistics*, **38**, 1034–1070.

Flegal, J. M. and Jones, G. L. (2011) Implementing Markov chain Monte Carlo: Estimating with confidence. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X., editors, *Handbook of Markov Chain Monte Carlo*, pages 175–197. Chapman & Hall/CRC Press.

Flegal, J. M., Jones, G. L., and Neath, R. (2012) Markov chain Monte Carlo estimation of quantiles. *University of California, Riverside, Technical Report*.

Gong, L., and Flegal, J. M. A practical sequential stopping rule for high-dimensional Markov chain Monte Carlo. Journal of Computational and Graphical Statistics (to appear).

Jones, G. L., Haran, M., Caffo, B. S. and Neath, R. (2006) Fixed-width output analysis for Markov chain Monte Carlo. *Journal of the American Statistical Association*, **101**, 1537–1547.

Vats, D., Flegal, J. M., and, Jones, G. L Multivariate Output Analysis for Markov chain Monte Carlo, *arXiv preprint arXiv:1512.07713 (2015)*.

See Also

mcse.mat, which applies mcse to each column of a matrix or data frame.

mcse.multi, for a multivariate estimate of the Monte Carlo standard error.

mcse.q and mcse.q.mat, which compute standard errors for quantiles.

Examples

```
# Create 10,000 iterations of an AR(1) Markov chain with rho = 0.9.

n = 10000
x = double(n)
x[1] = 2
for (i in 1:(n - 1))
    x[i + 1] = 0.9 * x[i] + rnorm(1)

# Estimate the mean, 0.1 quantile, and 0.9 quantile with MCSEs using batch means.

mcse(x)
mcse.q(x, 0.1)
mcse.q(x, 0.9)
```

```
# Estimate the mean, 0.1 quantile, and 0.9 quantile with MCSEs using overlapping batch means.

mcse(x, method = "obm")
mcse.q(x, 0.1, method = "obm")
mcse.q(x, 0.9, method = "obm")

# Estimate E(x^2) with MCSE using spectral methods.

g = function(x) { x^2 }
mcse(x, g = g, method = "tukey")
```

`mcse.initseq` *Multivariate Monte Carlo standard errors for expectations with the initial sequence method of Dai and Jones (2017).*

Description

Function returns the estimate of the covariance matrix in the Markov Chain CLT using initial sequence method. The function also returns the volume of the resulting ellipsoidal confidence regions. This method is designed to give an asymptotically conservative estimate of the Monte Carlo standard error.

Usage

```
mcse.initseq(x, adjust = FALSE, g = NULL, level = 0.95)
```

Arguments

`x`	a matrix or data frame of Markov chain output. Number of rows is the Monte Carlo sample size.
`adjust`	logical; if TRUE, an adjustment is made to increase slightly the eigenvalues of the initial sequence estimator. The default is FALSE.
`g`	a function that represents features of interest. g is applied to each row of `x` and thus g should take a vector input only. If g is `NULL`, g is set to be identity, which is estimation of the mean of the target density.
`level`	confidence level of the confidence region.

Value

A list is returned with the following components,

`cov`	a covariance matrix estimate using intial sequence method.
`cov.adj`	a covariance matrix estimate using adjusted initial sequence method if the input `adjust=TRUE`.
`vol`	volume of the confidence region to the pth root using intial sequence method, where p is the dimension of the confidence region.

vol.adj	volume of the confidence region to the pth root using adjusted initial sequence method if the input adjust=TRUE, where p is the dimension of the confidence region.
est	estimate of g(x).
nsim	number of rows of the input x.
adjust	logical of whether an adjustment was made to the initial sequence estimator.

References

Dai, N and Jones, G.L. (2017+) Multivariate initial sequence estimators in Markov chain Monte Carlo, *Journal of Multivariate Analysis*.

See Also

initseq{mcmc}, which is a different univariate initial sequence estimator. mcse, which acts on a vector. mcse.mat, which applies mcse to each column of a matrix or data frame. mcse.q and mcse.q.mat, which compute standard errors for quantiles. mcse.multi, which estimates the covariance matrix in the Markov Chain CLT using batch means or spectral variance methods.

Examples

```
library(mAr)
p <- 3
n <- 1000
omega <- 5*diag(1,p)

## Making correlation matrix var(1) model
set.seed(100)
foo <- matrix(rnorm(p^2), nrow = p)
foo <- foo %*% t(foo)
phi <- foo / (max(eigen(foo)$values) + 1)

dat <- as.matrix(mAr.sim(rep(0,p), phi, omega, N = n))

out.mcse <- mcse.initseq(x = dat)
out.mcse.adj  <- mcse.initseq(x = dat,adjust = TRUE)

# If we are only estimating the mean of the first component,
# and the second moment of the second component
g <- function(x) return(c(x[1], x[2]^2))
out.g.mcse <- mcse.initseq(x = dat, g = g)
```

---

mcse.mat *Apply* mcse *to each column of a matrix or data frame of MCMC samples.*

Description

Apply mcse to each column of a matrix or data frame of MCMC samples.

Usage

```
mcse.mat(x, size = "sqroot", g = NULL,
  method = c("bm", "obm", "tukey", "bartlett"))
```

Arguments

x	a matrix or data frame with each row being a draw from the multivariate distribution of interest.
size	the batch size. The default value is "sqroot", which uses the square root of the sample size. "cuberoot" will cause the function to use the cube root of the sample size. A numeric value may be provided if neither "sqroot" nor "cuberoot" is satisfactory.
g	a function such that $E(g(x))$ is the quantity of interest. The default is NULL, which causes the identity function to be used.
method	the method used to compute the standard error. This is one of "bm" (batch means, the default), "obm" (overlapping batch means), "tukey" (spectral variance method with a Tukey-Hanning window), or "bartlett" (spectral variance method with a Bartlett window).

Value

mcse.mat returns a matrix with ncol(x) rows and two columns. The row names of the matrix are the same as the column names of x. The column names of the matrix are "est" and "se". The jth row of the matrix contains the result of applying mcse to the jth column of x.

See Also

mcse, which acts on a vector. mcse.multi, for a multivariate estimate of the Monte Carlo standard error. mcse.q and mcse.q.mat, which compute standard errors for quantiles.

---

mcse.multi *Multivariate Monte Carlo standard errors for expectations.*

Description

Function returns the estimate of the covariance matrix in the Markov Chain CLT using batch means or spectral variance methods (with different lag windows). The function also returns the volume of the resulting ellipsoidal confidence regions.

Usage

```
mcse.multi(x, method = "bm", size = "sqroot", g = NULL, level = 0.95, large = FALSE)
```

Arguments

x	a matrix or data frame of Markov chain output. Number of rows is the Monte Carlo sample size.
method	any of ``bm'', ``bartlett'', ``tukey''. ``bm'' represents batch means estimator, ``bartlett'' and ``tukey'' represents the modified-Bartlett window and the Tukey-Hanning windows for the spectral variance estimators.
size	can take character values of ``sqroot'' and ``cuberoot'' or any numeric value between 1 and n. Size represents the batch size in bm and the truncation point in bartlett and tukey. sqroot means size is floor(n^(1/2) and cuberoot means size is floor(n^(1/3)).
g	a function that represents features of interest. g is applied to each row of x and thus g should take a vector input only. If g is NULL, g is set to be identity, which is estimation of the mean of the target density.
level	confidence level of the confidence ellipsoid.
large	if TRUE, returns the volume of the large sample confidence region using a chi square critical value.

Value

A list is returned with the following components,

cov	a covariance matrix estimate.
vol	volume of the confidence ellipsoid to the pth root.
est	estimate of g(x).
nsim	number of rows of the input x.
method	method used to calculate matrix cov.
large	logical of whether a large sample confidence region volume was calculated.
size	value of size used to calculate cov.

References

Vats, D., Flegal, J. M., and, Jones, G. L Multivariate Output Analysis for Markov chain Monte Carlo, *arXiv preprint arXiv:1512.07713 (2015)*.

Vats, D., Flegal, J. M., and, Jones, G. L. (2017+) Strong Consistency of multivariate spectral variance estimators for Markov chain Monte Carlo, *Bernoulli*.

See Also

mcse.initseq, which computes an initial sequence estimator. mcse, which acts on a vector. mcse.mat, which applies mcse to each column of a matrix or data frame. mcse.q and mcse.q.mat, which compute standard errors for quantiles.

Examples

```
library(mAr)
p <- 3
n <- 1e3
omega <- 5*diag(1,p)

## Making correlation matrix var(1) model
set.seed(100)
foo <- matrix(rnorm(p^2), nrow = p)
foo <- foo %*% t(foo)
phi <- foo / (max(eigen(foo)$values) + 1)

out <- as.matrix(mAr.sim(rep(0,p), phi, omega, N = n))

mcse.bm <- mcse.multi(x = out)
mcse.tuk <- mcse.multi(x = out, method = "tukey")

# If we are only estimating the mean of the first component,
# and the second moment of the second component

g <- function(x) return(c(x[1], x[2]^2))
mcse <- mcse.multi(x = out, g = g)
```

`mcse.q` *Compute Monte Carlo standard errors for quantiles.*

Description

Compute Monte Carlo standard errors for quantiles.

Usage

```
mcse.q(x, q, size = "sqroot", g = NULL,
  method = c("bm", "obm", "sub"), warn = FALSE)
```

Arguments

`x`	a vector of values from a Markov chain.
`q`	the quantile of interest.
`size`	the batch size. The default value is "`sqroot`", which uses the square root of the sample size. A numeric value may be provided if "`sqroot`" is not satisfactory.
`g`	a function such that the qth quantile of the univariate distribution function of $g(x)$ is the quantity of interest. The default is `NULL`, which causes the identity function to be used.
`method`	the method used to compute the standard error. This is one of "`bm`" (batch means, the default), "`obm`" (overlapping batch means), or "`sub`" (subsampling bootstrap).
`warn`	a logical value indicating whether the function should issue a warning if the sample size is too small (less than 1,000).

Value

mcse.q returns a list with two elements:

est	an estimate of the qth quantile of the univariate distribution function of $g(x)$.
se	the Monte Carlo standard error.

References

Flegal, J. M. (2012) Applicability of subsampling bootstrap methods in Markov chain Monte Carlo. In Wozniakowski, H. and Plaskota, L., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010* (to appear). Springer-Verlag.

Flegal, J. M. and Jones, G. L. (2010) Batch means and spectral variance estimators in Markov chain Monte Carlo. *The Annals of Statistics*, **38**, 1034–1070.

Flegal, J. M. and Jones, G. L. (2011) Implementing Markov chain Monte Carlo: Estimating with confidence. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X., editors, *Handbook of Markov Chain Monte Carlo*, pages 175–197. Chapman & Hall/CRC Press.

Flegal, J. M., Jones, G. L., and Neath, R. (2012) Markov chain Monte Carlo estimation of quantiles. *University of California, Riverside, Technical Report*.

Jones, G. L., Haran, M., Caffo, B. S. and Neath, R. (2006) Fixed-width output analysis for Markov chain Monte Carlo. *Journal of the American Statistical Association*, **101**, 1537–1547.

See Also

mcse.q.mat, which applies mcse.q to each column of a matrix or data frame.

mcse and mcse.mat, which compute standard errors for expectations.

Examples

```
# Create 10,000 iterations of an AR(1) Markov chain with rho = 0.9.

n = 10000
x = double(n)
x[1] = 2
for (i in 1:(n - 1))
    x[i + 1] = 0.9 * x[i] + rnorm(1)

# Estimate the mean, 0.1 quantile, and 0.9 quantile with MCSEs using batch means.

mcse(x)
mcse.q(x, 0.1)
mcse.q(x, 0.9)

# Estimate the mean, 0.1 quantile, and 0.9 quantile with MCSEs using overlapping batch means.

mcse(x, method = "obm")
mcse.q(x, 0.1, method = "obm")
mcse.q(x, 0.9, method = "obm")

# Estimate E(x^2) with MCSE using spectral methods.
```

```
g = function(x) { x^2 }
mcse(x, g = g, method = "tukey")
```

`mcse.q.mat` *Apply* `mcse.q` *to each column of a matrix or data frame of MCMC samples.*

Description

Apply `mcse.q` to each column of a matrix or data frame of MCMC samples.

Usage

```
mcse.q.mat(x, q, size = "sqroot", g = NULL,
  method = c("bm", "obm", "sub"))
```

Arguments

`x`	a matrix or data frame of Markov chain output. Number of rows is the Monte Carlo sample size.
`q`	the quantile of interest.
`size`	the batch size. The default value is "`sqroot`", which uses the square root of the sample size. "`cuberoot`" will cause the function to use the cube root of the sample size. A numeric value may be provided if "`sqroot`" is not satisfactory.
`g`	a function such that the qth quantile of the univariate distribution function of $g(x)$ is the quantity of interest. The default is `NULL`, which causes the identity function to be used.
`method`	the method used to compute the standard error. This is one of "`bm`" (batch means, the default), "`obm`" (overlapping batch means), or "`sub`" (subsampling bootstrap).

Value

`mcse.q.mat` returns a matrix with `ncol(x)` rows and two columns. The row names of the matrix are the same as the column names of `x`. The column names of the matrix are "`est`" and "`se`". The jth row of the matrix contains the result of applying `mcse.q` to the jth column of `x`.

See Also

`mcse.q`, which acts on a vector.

`mcse` and `mcse.mat`, which compute standard errors for expectations.

`minESS` *Minimum effective sample size required for stable estimation as described in Vats et al. (2015).*

Description

The function calculates the minimum effective sample size required for a specified relative tolerance level. This function can also calculate the relative precision in estimation for a given estimated effective sample size.

Usage

```
minESS(p, alpha = .05, eps = .05, ess = NULL)
```

Arguments

`p`	dimension of the estimation problem.
`alpha`	confidence level
`eps`	tolerance level. The `eps` value is ignored is `ess` is not `NULL`
`ess`	Estimated effective sample size. Usually the output value from `multiESS`.

Details

The minimum effective samples required when estimating a vector of length p, with 100(1-α)% confidence and tolerance of ϵ is

$$\text{mESS} \geq \frac{2^{2/p}\pi}{(p\Gamma(p/2))^{2/p}} \frac{\chi^2_{1-\alpha,p}}{\epsilon^2}$$

The above equality can also be used to get ϵ from an already obtained estimate of mESS.

Value

By default function returns the minimum effective sample required for a given `eps` tolerance. If `ess` is specified, then the value returned is the `eps` corresponding to that `ess`.

References

Gong, L., and Flegal, J. M. A practical sequential stopping rule for high-dimensional Markov chain Monte Carlo. Journal of Computational and Graphical Statistics (to appear).

Vats, D., Flegal, J. M., and, Jones, G. L Multivariate Output Analysis for Markov chain Monte Carlo, *arXiv preprint arXiv:1512.07713 (2015)*.

See Also

`multiESS`, which calculates multivariate effective sample size using a Markov chain and a function g.

`ess` which calculates univariate effective sample size using a Markov chain and a function g.

Examples

```
minESS(p = 5)
```

`multiESS` *Effective Sample Size of a multivariate Markov chain as described in Vats et al. (2015).*

Description

Calculate the effective sample size of the Markov chain, using the multivariate dependence structure of the process.

Usage

```
multiESS(x, covmat = NULL, g = NULL, ...)
```

Arguments

`x`	a matrix or data frame of Markov chain output. Number of rows is the Monte Carlo sample size.
`covmat`	optional matrix estimate obtained using `mcse.multi` or `mcse.initseq`.
`g`	a function that represents features of interest. g is applied to each row of `x` and thus g should take a vector input only. If g is `NULL`, g is set to be identity, which is estimation of the mean of the target density.
`...`	arguments for `mcse.multi` function. Don't use this if a suitable matrix estimate from `mcse.multi` or `mcse.initseq` is already obtained. .

Details

Effective sample size is the size of an iid sample with the same variance as the current sample. ESS is given by

$$\text{ESS} = n\frac{|\Lambda|^{1/p}}{|\Sigma|^{1/p},}$$

where Λ is the sample covariance matrix for g and Σ is an estimate of the Monte Carlo standard error for g.

Value

The function returns the estimated effective sample size.

References

Vats, D., Flegal, J. M., and, Jones, G. L Multivariate Output Analysis for Markov chain Monte Carlo, *arXiv preprint arXiv:1512.07713 (2015).*

See Also

minESS, which calculates the minimum effective samples required for the problem.

ess which calculates univariate effective sample size using a Markov chain and a function g.

Examples

```
library(mAr)
p <- 3
n <- 1e3
omega <- 5*diag(1,p)

## Making correlation matrix var(1) model
set.seed(100)
foo <- matrix(rnorm(p^2), nrow = p)
foo <- foo %*% t(foo)
phi <- foo / (max(eigen(foo)$values) + 1)

out <- as.matrix(mAr.sim(rep(0,p), phi, omega, N = n))

multiESS(out)
```

---

qqTest *QQplot for Markov chains*

Description

QQplot for Markov chains using an estimate of the Markov Chain CLT covariance matrix.

Usage

```
qqTest(mcse.obj)
```

Arguments

mcse.obj	the list returned by the mcse.multi or mcse.initseq command

Examples

```
library(mAr)
p <- 35
n <- 1e4
omega <- 5*diag(1,p)

## Making correlation matrix var(1) model
set.seed(100)
foo <- matrix(rnorm(p^2), nrow = p)
foo <- foo %*% t(foo)
phi <- foo / (max(eigen(foo)$values) + 1)
```

```
out <- as.matrix(mAr.sim(rep(0,p), phi, omega, N = n))

mcse.bm <- mcse.multi(x = out)
qqTest(mcse.bm)

mcse.isadj <- mcse.initseq(x = out, adjust = TRUE)
qqTest(mcse.isadj)
```

Index