Package 'stableGR'

July 23, 2025

Type Package

Title A Stable Gelman-Rubin Diagnostic for Markov Chain Monte Carlo

Version 1.2

Date 2022-10-7

Maintainer Christina Knudson <drchristinaknudson@gmail.com>

Description Practitioners of Bayesian statistics often use Markov chain Monte Carlo (MCMC) samplers to sample from a posterior distribution. This package determines whether the MCMC sample is large enough to yield reliable estimates of the target distribution. In particular, this calculates a Gelman-Rubin convergence diagnostic using stable and consistent estimators of Monte Carlo variance. Additionally, this uses the connection between an MCMC sample's effective sample size and the Gelman-Rubin diagnostic to produce a threshold for terminating MCMC simulation. Finally, this informs the user whether enough samples have been collected and (if necessary) estimates the number of samples needed for a desired level of accuracy. The theory underlying these methods can be found in ``Revisiting the Gelman-Rubin Diagnostic'' by Vats and Knudson (2018) <arXiv:1812:09384>.

License GPL-3

Depends R (>= 3.5), mcmcse(>= 1.4-1)

Imports mvtnorm

ByteCompile TRUE

Repository CRAN

RoxygenNote 7.1.1

NeedsCompilation no

Author Christina Knudson [aut, cre],
Dootika Vats [aut]

Date/Publication 2022-10-07 22:50:02 UTC

Contents

---

stableGR-package   *A Stable Gelman-Rubin Diagnostic for Markov Chain Monte Carlo*

Description

Practitioners of Bayesian statistics often use Markov chain Monte Carlo (MCMC) samplers to sample from a posterior distribution. This package determines whether the MCMC sample is large enough to yield reliable estimates of the target distribution. In particular, this calculates a Gelman-Rubin convergence diagnostic using stable and consistent estimators of Monte Carlo variance. Additionally, this uses the connection between an MCMC sample's effective sample size and the Gelman-Rubin diagnostic to produce a threshold for terminating MCMC simulation. Finally, this informs the user whether enough samples have been collected and (if necessary) estimates the number of samples needed for a desired level of accuracy. The theory underlying these methods can be found in "Revisiting the Gelman-Rubin Diagnostic" by Vats and Knudson (2018) <arXiv:1812:09384>.

Details

Package:	stableGR
Type:	Package
Title:	A Stable Gelman-Rubin Diagnostic for Markov Chain Monte Carlo
Version:	1.2
Date:	2022-10-7
Authors@R:	c(person("Christina", "Knudson", role = c("aut", "cre"), email = "drchristinaknudson@gmail.com"), person("
Maintainer:	Christina Knudson <drchristinaknudson@gmail.com>
Description:	Practitioners of Bayesian statistics often use Markov chain Monte Carlo (MCMC) samplers to sample from a
License:	GPL-3
Depends:	R (>= 3.5), mcmcse(>= 1.4-1)
Imports:	mvtnorm
ByteCompile:	TRUE
Repository:	CRAN
RoxygenNote:	7.1.1
Author:	Christina Knudson [aut, cre], Dootika Vats [aut]

Practitioners of Bayesian statistics often use Markov chain Monte Carlo (MCMC) samplers to sample from a posterior distribution. This package determines whether the MCMC sample is large enough to yield reliable estimates of the target distribution. In particular, this calculates a Gelman-Rubin convergence diagnostic using stable and consistent estimators of Monte Carlo variance. Additionally, this uses the connection between an MCMC sample's effective sample size

and the Gelman-Rubin diagnostic to produce a threshold for terminating MCMC simulation. Finally, this informs the user whether enough samples have been collected and (if necessary) estimates the number of samples needed for a desired level of accuracy. The theory underlying these methods can be found in "Revisiting the Gelman-Rubin Diagnostic" by Vats and Knudson (2018) <arXiv:1812:09384>.

This package is unique in a few ways. First, it uses stable variance estimators to calculate a stabilized Gelman-Rubin statistic. Second, it leverages the connection between effective sample size and the potential scale reduction factor (PSRF). Third, this diagnostic can be used whether MCMC samples were created from a single chain or multiple chains.

The main functions in the package are `stable.GR`, `n.eff`, and `target.psrf`. `stable.GR` returns the univariate PSRF, the multivariate PSRF, and the estimated effective sample size. `n.eff` returns informs the user whether sufficient MCMC samples have been collected; if not, `n.eff` also returns the estimated target sample size `target.psrf` creates a termination threshold for `stable.GR`; MCMC sampling can terminate when the MCMC samples' psrf is smaller than the value returned by `target.psrf`.

Author(s)

NA

Maintainer: Christina Knudson <drchristinaknudson@gmail.com>

References

Vats, D. and Knudson, C. Revisiting the Gelman-Rubin Diagnostic. arXiv:1812.09384.

Vats, D. and Flegal, J. Lugsail lag windows and their application to MCMC. arXiv: 1809.04541.

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.

Gelman, A and Rubin, DB (1992) Inference from iterative simulation using multiple sequences, *Statistical Science*, **7**, 457-511.

Brooks, SP. and Gelman, A. (1998) General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**, 434-455.

Examples

```
library(stableGR)
set.seed(100)
p <- 2
n <- 100 # For real problems, use a MUCH larger n.

sig.mat = matrix(c(1, .8, .8, 1), ncol = 2, nrow = 2)

# Making 3 chains
chain1 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain2 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain3 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)

# find GR diagnostic using all three chains
x <- list(chain1, chain2, chain3)
```

```
stable.GR(x)
```

---

asym.var   *Asymptotic covariance matrix estimation for Markov chain Monte*
                             Carlo

Description

Estimates the asymptotic covariance matrix for Monte Carlo estimators, compatible with multiple
chains. If a single chain is input, it calls `mcmcse::mcse.multi`.

Usage

```
asym.var(
  x,
  multivariate = TRUE,
  method = "lug",
  size = NULL,
  autoburnin = FALSE,
  adjust = TRUE
)
```

Arguments

x	a list of matrices, where each matrix is $n \times p$. Each row of the matrices represents one step of the chain. Each column of the matrices represents one variable. A list with a single matrix (chain) is allowed. Optionally, this can be an `mcmclist` object.
multivariate	a logical flag indicating whether the full matrix is returned (TRUE) or only the diagonals (FALSE)
method	the method used to compute the matrix. This is one of "`lug`" (lugsail, the default), "`bm`" (batch means), "`obm`" (overlapping batch means), "`tukey`" (spectral variance method with a Tukey-Hanning window), or "`bartlett`" (spectral variance method with a Bartlett window).
size	options are NULL (default, which calculates an ideal batch size), character values of `sqroot` and `cuberoot`, or any numeric value between 1 and n. Size represents the batch size in "`bm`" (batch means) 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)).
autoburnin	a logical flag indicating whether only the second half of the series should be used in the computation. If set to TRUE and `start(x)` is less than `end(x)/2` then start of series will be adjusted so that only second half of series is used.
adjust	this argument is now obselete due to package updates.

Details

The function returns estimate of the univariate or multivariate asymptotic (co)variance of Monte Carlo estimators. If $X_1, \ldots X_n$ are the MCMC samples, then function returns the estimate of $\lim_{n \to \infty} nVar(\bar{X})$. In other words, if a Markov chain central limit holds such that, as $n \to \infty$

$$\sqrt{n}(\bar{X} - \mu) \to N(0, \Sigma)$$

then the function returns an estimator of Σ from the m different chains. If `multivariate == FALSE`, then only the diagonal of Σ are returned.

Value

The asymptotic variance estimate (if `multivariate = FALSE`) or the asymptotic covariance matrix (if `multivariate = TRUE`) in the Markov chain central limit theorem.

References

Vats, D. and Knudson, C. Revisiting the Gelman-Rubin Diagnostic. arXiv:1812.09384.

Vats, D. and Flegal, J. Lugsail lag windows and their application to MCMC. arXiv: 1809.04541.

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.

Gelman, A and Rubin, DB (1992) Inference from iterative simulation using multiple sequences, *Statistical Science*, **7**, 457-511.

Brooks, SP. and Gelman, A. (1998) General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**, 434-455.

Examples

```
library(stableGR)
set.seed(100)
p <- 2
n <- 100 # n is tiny here purely for demo purposes.
# use n much larger for real problems!

sig.mat = matrix(c(1, .8, .8, 1), ncol = 2, nrow = 2)

# Making 3 chains
chain1 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain2 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain3 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)

# find GR diagnostic using all three chains
x <- list(chain1, chain2, chain3)
asym.var(x)
```

`mvn.gibbs` *Two block Gibbs sampler for a multivariate normal distribution*

Description

This function generates a Markov chain sample from a multivariate normal distribution using a two-block Gibbs sampler. The function is used mainly for implementation in the examples of this package.

Usage

```
mvn.gibbs(N = 10000, p, mu, sigma)
```

Arguments

N	number of Markov chain samples desired
p	dimension of the multivariate normal target distribution
mu	mean vector of the multivariate normal distribution
sigma	covariance matrix of the multivariate normal distribution

Value

N by p matrix of samples from the multivariate normal target distribution

---

`n.eff` *Effective sample size*

Description

For an estimator, effective sample size is the number of independent samples with the same standard error as a correlated sample. This function calculates effective sample size for a set of Markov chains using lugsail variance estimators. This also determines whether the Markov chains have converged. If they have not, this function approximates the number of samples needed.

Usage

```
n.eff(
  x,
  multivariate = TRUE,
  epsilon = 0.05,
  delta = NULL,
  alpha = 0.05,
  method = "lug",
  size = NULL,
  autoburnin = FALSE
)
```

Arguments

x	a list of matrices, where each matrix represents one Markov chain sample. Each row of the matrices represents one step of the chain. Each column of the matrices represents one variable. A list with a single matrix (chain) is allowed. Optionally, this can be an `mcmclist` object.
multivariate	a logical flag indicating whether the effective sample size should be calculated for multivariate chains.
epsilon	relative precision level. Values less than .10 are recommended.
delta	desired delta value - the cutoff for potential scale reduction factor.
alpha	significance level for confidence regions for the Monte Carlo estimators.
method	the method used to compute the standard error of the chains. This is one of "`lug`" (lugsail, the default), "`bm`" (batch means), "`obm`" (overlapping batch means), "`tukey`" (spectral variance method with a Tukey-Hanning window), or "`bartlett`" (spectral variance method with a Bartlett window).
size	options are NULL (default, which calculates an ideal batch size), character values of sqroot and cuberoot, or any numeric value between 1 and n. Size represents the batch size in "`bm`" (batch means) 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)).
autoburnin	a logical flag indicating whether only the second half of the series should be used in the computation. If set to TRUE and start(x) is less than end(x)/2 then start of series will be adjusted so that only second half of series is used.

Value

n.eff	a scalar point estimate of the effective sample size.
converged	a logical indicating whether sufficient samples have been obtained.
n.target	NULL (if converged == TRUE) or a scalar estimate of the chain length required for convergence, assuming the number of chains is unchanged.

References

Vats, D. and Knudson, C. Revisiting the Gelman-Rubin Diagnostic. arXiv:1812.09384.

Vats, D. and Flegal, J. Lugsail lag windows and their application to MCMC. arXiv: 1809.04541.

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.

Gelman, A and Rubin, DB (1992) Inference from iterative simulation using multiple sequences, *Statistical Science*, **7**, 457-511.

Brooks, SP. and Gelman, A. (1998) General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**, 434-455.

Examples

```
library(stableGR)
set.seed(100)
p <- 2
n <- 100 # n is tiny here purely for demo purposes.
# use n much larger for real problems!


sig.mat = matrix(c(1, .8, .8, 1), ncol = 2, nrow = 2)
# Making 3 chains
chain1 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain2 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain3 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)

# find ESS using all three chains
x <- list(chain1, chain2, chain3)
n.eff(x)
```

---

stable.GR   *Gelman-Rubin diagnostic using stable variance estimators*

Description

This function uses fast and strongly consistent estimators estimators of Monte Carlo variance to calculate the Gelman-Rubin convergence diagnostic for Markov chain Monte Carlo. A univariate 'potential scale reduction factor' (PSRF) is calculated for each variable in x. For multivariate chains, a multivariate PSRF is calculated to take into account the interdependence of the chain's components. The PSRFs decrease to 1 as the chain length increases. When the PSRF becomes sufficiently close to 1, the sample collected by the Markov chain has converged to the target distribution.

Usage

```
stable.GR(
  x,
  multivariate = TRUE,
  mapping = "determinant",
  method = "lug",
  size = NULL,
  autoburnin = FALSE,
  blather = FALSE
)
```

Arguments

x	a list of matrices, where each matrix represents one Markov chain sample. Each row of the matrices represents one step of the chain. Each column of the matrices represents one variable. A list with a single matrix (chain) is allowed. Optionally, this can be an mcmclist object.
multivariate	a logical flag indicating whether the multivariate potential scale reduction factor should be calculated for multivariate chains.
mapping	the function used to map the covariance matrix to a scalar. This is one of "determinant" (determinant of the covariance matrix, the default) or "maxeigen" (the largest eigenvalue of the covariance matrix).
method	the method used to compute the standard error of the chains. This is one of "lug" (lugsail, the default), "bm" (batch means), "obm" (overlapping batch means), "tukey" (spectral variance method with a Tukey-Hanning window), or "bartlett" (spectral variance method with a Bartlett window).
size	options are NULL (default, which calculates an ideal batch size), character values of sqroot and cuberoot, or any numeric value between 1 and n. Size represents the batch size in "bm" (batch means) 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)).
autoburnin	a logical flag indicating whether only the second half of the series should be used in the computation. If set to TRUE and start(x) is less than end(x)/2 then start of series will be adjusted so that only second half of series is used.
blather	a logical flag indicating whether to include additional output.

Value

psrf	A vector containing the point estimates of the PSRF.
mpsrf	A scalar point estimate of the multivariate PSRF.
means	A vector containing the sample means based on the chains provided.
n.eff	A scalar point estimate of the effective sample size.
blather	Either FALSE or a list containing intermediate calculations.

Theory

Gelman and Rubin (1992) and Brooks and Gelman (1998) first constructed the univariate and multivariate potential scale reduction factors (PSRF), respectively, to diagnose Markov chain convergence. The function stable.GR stabilizes the PSRF and improves the PSRF's efficiency by incorporating lugsail estimators for the target variance. The PSRF decreases to 1 as the chain length increases; when the PSRF becomes sufficiently close to 1, the sample collected by the Markov chain has converged to to the target distribution. A PSRF convergence threshold can be calculated using choosepsrf.

References

Vats, D. and Knudson, C. Revisiting the Gelman-Rubin Diagnostic. arXiv:1812.09384.

Vats, D. and Flegal, J. Lugsail lag windows and their application to MCMC. arXiv: 1809.04541.

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.

Gelman, A and Rubin, DB (1992) Inference from iterative simulation using multiple sequences, *Statistical Science*, **7**, 457-511.

Brooks, SP. and Gelman, A. (1998) General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**, 434-455.

Examples

```
library(stableGR)
set.seed(100)
p <- 2
n <- 100 # n is tiny here purely for demo purposes.
# use n much larger for real problems!

sig.mat = matrix(c(1, .8, .8, 1), ncol = 2, nrow = 2)

# Making 3 chains
chain1 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain2 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)
chain3 <- mvn.gibbs(N = n, p = p, mu = rep(1,p), sigma = sig.mat)

# find GR diagnostic using all three chains
x <- list(chain1, chain2, chain3)
stable.GR(x)
```

| target.psrf | *Calculates a Gelman Rubin diagnostic threshold using effective sample size thresholds.* |

Description

When the sample diagnostic reaches the psrf threshold calculated in this function, sufficient samples have been obtained.

Usage

```
target.psrf(p, m, epsilon = 0.05, delta = NULL, alpha = 0.05)
```

Arguments

p	dimension of the estimation problem.
m	number of chains.
epsilon	relative precision level. Values less than .10 are recommended.
delta	desired delta value - the cutoff for potential scale reduction factor. If specified, then the corresponding epsilon is returned.
alpha	significance level for confidence regions for the Monte Carlo estimators.

Value

psrf	The desired PSRF cutoff to stop the simulation.
epsilon	The epsilon value used to calculate the PSRF threshold.

References

Vats, D. and Knudson, C. Revisiting the Gelman-Rubin Diagnostic. arXiv:1812.09384

Vats, D. and Flegal, J. Lugsail lag windows and their application to MCMC. arXiv: 1809.04541.

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.

Gelman, A and Rubin, DB (1992) Inference from iterative simulation using multiple sequences, *Statistical Science*, **7**, 457-511.

Brooks, SP. and Gelman, A. (1998) General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**, 434-455.

Examples

```
target.psrf(p = 2, m = 3, epsilon = .05,  alpha = .05)
```

---

titanic.complete	*Titanic passenger data*

Description

Titanic passenger survival data. Complete cases only.

Usage

```
data(titanic.complete)
```

Format

A data frame with the following columns:

Survived Whether a passenger survived.

Pclass The class of the passenger's ticket. A factor with 3 levels.

Sex Male or female. A factor with 2 levels.

Age The age of the passenger.

SibSp The number of siblings/spouse aboard.

Parch The number of parents/children aboard.

Fare The passenger's fare.

Embarked The passenger's port of embarkation. A factor with 3 levels.

References

https://www.kaggle.com/c/titanic/data

Examples

```
data(titanic.complete)
```

Index