Package ‘bmk’

February 19, 2015

Type Package

Title MCMC diagnostics package

Version 1.0

Date 2012-09-21

Author Matthew Krachey and Edward L. Boone

Maintainer Matthew Krachey <matthewkrachey@yahoo.com>

Description MCMC diagnostic package that contains tools to diagnose convergence as well as to evaluate sensitivity studies, Includes summary functions which output mean, median, 95percentCI, Gelman & Rubin diagnostics and the Hellinger distance based diagnostics, Also contains functions to determine when an MCMC chain has converged via Hellinger distance, A function is also provided to compare outputs from identically dimensioned chains for determining sensitivy to prior distribution assumptions

License GPL (>= 2)

Depends coda, plyr, functional

Collate 'bmkconverge.R' 'bmk.R' 'bmksensitive.R' 'bmksummary.R' 'HBconverg1.R' 'HDistNoSize.R' 'HDistSize.R' 'HWconverg1.R'

Repository CRAN

Date/Publication 2012-10-17 07:29:42

NeedsCompilation no

R topics documented:

bmkconverge *bmkcoverge: Convergence via the Hellinger distance*

Description

MCMC chain convergence diagnostic.

Usage

```
bmkconverge(inputlist1, binsize = 1000)
```

Arguments

inputlist1	A list of the MCMC chains
binsize	a scalar giving how large each bin should be for consecutive batches. outputs the Hellinger distances between the sampled distribution for one scenario against the other.

Details

This takes an MCMC chain and divides it into batches of size binsize and calculates the Hellinger distance between consecutive batches.

References

Boone EL, Merrick JR and Krachey MJ. A Hellinger distance approach to MCMC diagnostics. Journal of Statistical Computation and Simulation, DOI:10.1080/00949655.2012.729588.

Examples

```
## Not run:
library(dismo); library(MCMCpack);
data(Anguilla_train)
b0mean <- 0
b0precision <- (1/5)^2
mcmclen = 1000
burn=10000
MCMC.one <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                b0=b0mean, B0=b0precision)
```

```
## End(Not run)
 data(MCMCsamples)
 mcmclen <- 1000
 bmkconverge(MCMC.one,mcmclen/10)
```

bmksensitive *Hellinger distance between two MCMC chains for sensitivity studies*

Description

Determine if two identically dimensioned sets of chains match. This is good for conducting sensitivity studies.

Usage

```
bmksensitive(inputlist1, inputlist2)
```

Arguments

`inputlist1`	A list of the combined MCMC chains for all samples from one scenario.
`inputlist2`	A list of the combined MCMC chains for all samples from another scenario.

References

Boone EL, Merrick JR and Krachey MJ. A Hellinger distance approach to MCMC diagnostics. Journal of Statistical Computation and Simulation, DOI:10.1080/00949655.2012.729588.

Examples

```
data(MCMCsamples)
 bmksensitive(MCMC.one.mean0, MCMC.one.mean1)
 ## Not run:
 library(dismo); library(MCMCpack)
 data(Anguilla_train)
 b0mean0 <- 0
 b0mean1 <- 1
 b0precision <- (1/5)^2
 mcmclen = 1000
 burn=10000
 MCMC.one.mean0 <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                  b0=b0mean0, B0=b0precision)
 MCMC.one.mean1 <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-.5,
                  b0=b0mean1, B0=b0precision)
 bmksensitive(one, two)

## End(Not run)
```

bmksummary *Summaries of MCMC chains*

Description

Generate the mean, standard deviation, median, 2.5 percent and 97.5 percent quantiles, Gelman Rubin statistic for convergence, effective samples size and the minimum and maximum Hellinger distances across all chains. outputs summaries for the MCMC samples including the convergence diagnostics of Gelman and Rubin and the Hellinger distance of Boone, Merrick and Krachey.

Usage

```
bmksummary(inputlist)
```

Arguments

`inputlist`	A list of the combined MCMC chains for all samples from one scenario.

References

Boone EL, Merrick JR and Krachey MJ. A Hellinger distance approach to MCMC diagnostics. Journal of Statistical Computation and Simulation, `DOI:10.1080/00949655.2012.729588`.

Examples

```
data(MCMCsamples)
 bmksummary(list( MCMC.one, MCMC.two, MCMC.three ))
```

HBconverg1 *Hellinger distance between distributions*

Description

This computes the Hellinger distance for all pairwise combinations of MCMC chains.

Usage

```
HBconverg1(chains1)
```

Arguments

`chains1`	A matrix of MCMC for the same variable. Each column corresponds to a different chain.

Value

c2 A vector containing the minimum and maximum Hellinger distances across all pairwise comparisons.

Note

The matrix must consist of samples from the same variable derived from different chains.

HDistNoSize — *Hellinger distance between two MCMC chains using default grid in kernel density estimator.*

Description

This function determines the Hellinger distance between two MCMC chains via kernel density estimates.

Usage

```
HDistNoSize(b1, b2)
```

Arguments

`b1`	vector of first MCMC chain.
`b2`	vector of second MCMC chain.

Value

The Hellinger distance between the kernel density estimates for b1 and b2.

Note

The chains need to be the same length.

HDistSize — *Hellinger distance between two MCMC chains using a specified grid size.*

Description

This function determines the Hellinger distance between two MCMC chains via kernel density estimates.

Usage

```
HDistSize(b1, b2, n2)
```

Arguments

`b1`	vector of first MCMC chain.
`b2`	vector of second MCMC chain.
`n2`	is the number of divisions to run for the kernel density estimator

Value

res1 The Hellinger distance between the kernel density estimates for b1 and b2.

Note

The chains need to be the same length.

HWconverg1 — *Hellinger distance within consecutive batches of MCMC samples.*

Description

Determine if a specific chain has converged. This takes a chain and divides it into batches and calculates the Hellinger distance between consecutive batches.

Usage

```
HWconverg1(chain1, batchsize1 = 1000)
```

Arguments

`chain1`	A vector of a single MCMC chain.
`batchsize1`	An integer that defines the size of the batches.

Value

c2 A vector of Hellinger distances between consecutive batches.

MCMC.one *MCMC.one is an mcmc object resulting from the following code:*

Description

MCMC.one is an mcmc object resulting from the following code:

Author(s)

Edward L. Boone <elboone@vcu.edu>

Examples

```
## Not run:
 library(dismo); library(MCMCpack)
 data(Anguilla_train)
 b0mean0 <- 0
 b0precision <- (1/5)^2
 mcmclen = 50000
 burn=200000
 MCMC.one <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                  b0=b0mean0, B0=b0precision)

## End(Not run)
```

MCMC.one.mean0 *MCMC.one.mean0 is an mcmc object resulting from the following code:*

Description

MCMC.one.mean0 is an mcmc object resulting from the following code:

Author(s)

Edward L. Boone <elboone@vcu.edu>

Examples

```
## Not run:
 library(dismo); library(MCMCpack)
 data(Anguilla_train)
 b0mean0 <- 0
 b0precision <- (1/5)^2
 mcmclen = 50000
 burn=200000
 MCMC.one.mean0 <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                  b0=b0mean0, B0=b0precision)

## End(Not run)
```

MCMC.one.mean1 — *MCMC.one.mean1 is an mcmc object resulting from the following code:*

Description

MCMC.one.mean1 is an mcmc object resulting from the following code:

Author(s)

Edward L. Boone <elboone@vcu.edu>

Examples

```
## Not run:
 library(dismo); library(MCMCpack)
 data(Anguilla_train)
 b0mean1 <- 1
 b0precision <- (1/5)^2
 mcmclen = 50000
 burn=200000
 MCMC.one.mean1 <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                  b0=b0mean1, B0=b0precision)

## End(Not run)
```

MCMC.three — *MCMC.three is an mcmc object resulting from the following code:*

Description

MCMC.three is an mcmc object resulting from the following code:

Author(s)

Edward L. Boone <elboone@vcu.edu>

Examples

```
## Not run:
 library(dismo); library(MCMCpack)
 data(Anguilla_train)
 b0mean0 <- 0
 b0precision <- (1/5)^2
 mcmclen = 50000
 burn=200000
 MCMC.three <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                  b0=b0mean0, B0=b0precision)

## End(Not run)
```

MCMC.two *MCMC.two is an mcmc object resulting from the following code:*

Description

MCMC.two is an mcmc object resulting from the following code:

Author(s)

Edward L. Boone <elboone@vcu.edu>

Examples

```
## Not run:
 library(dismo); library(MCMCpack)
 data(Anguilla_train)
 b0mean0 <- 0
 b0precision <- (1/5)^2
 mcmclen = 50000
 burn=200000
 MCMC.two <- MCMClogit(Angaus ~ SegSumT+DSDist+USNative+as.factor(Method)+DSMaxSlope+USSlope,
                  data=Anguilla_train,burnin=burn, mcmc=mcmclen, beta.start=-1,
                  b0=b0mean0, B0=b0precision)

## End(Not run)
```

Index