

Package ‘mcmcr’

April 13, 2018

Title Manipulate MCMC Samples

Version 0.0.1

Description Functions and classes to store, manipulate and summarise Monte Carlo Markov Chain (MCMC) samples.

License MIT + file LICENSE

Depends R (>= 3.4.0)

Imports abind, checkr, coda, graphics, stats, tibble, utils

Suggests covr, testthat

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

NeedsCompilation no

Author Joe Thorley [aut, cre]

Maintainer Joe Thorley <joe@poissonconsulting.ca>

Repository CRAN

Date/Publication 2018-04-13 08:21:31 UTC

R topics documented:

anyNA	3
as.marray	4
as.mcmc	4
as.mcmc.list	5
as.mcmcarray	6
as.mcmcr	7
as.mcmcrs	8
as.term	8
bind_chains	9
bind_dimensions	10
bind_dimensions_n	11
bind_iterations	12

bind_parameters	13
coef	14
collapse_chains	15
combine_dimensions	16
combine_samples	17
combine_samples_n	18
converged	18
dims	19
esr	20
ess	21
estimates	22
is.marray	23
is.mcmarray	23
is.mcmcr	24
is.mcmcrs	24
is.term	25
mcmarray-object	25
mcmcr-object	26
mcmcrs	26
mcmcrs-object	27
mcmcr_example	27
mcmc_map	28
nchains	29
ndims	30
niters	31
npars	31
nsams	32
nsims	33
nterms	33
parameters	34
pdims	35
pvalue	36
rhat	37
sort	38
split_chains	39
subset	40
tdims	41
term	41
terms	42
thin	43
zero	44

anyNA

Any Missing Values

Description

Test whether there are any missing values.

Usage

```
## S3 method for class 'mccarray'  
anyNA(x, recursive = FALSE)  
  
## S3 method for class 'mcmc'  
anyNA(x, recursive = FALSE)  
  
## S3 method for class 'mcmc.list'  
anyNA(x, recursive = FALSE)  
  
## S3 method for class 'mcmcarray'  
anyNA(x, recursive = FALSE)  
  
## S3 method for class 'mcmcr'  
anyNA(x, recursive = FALSE)  
  
## S3 method for class 'mcmcrs'  
anyNA(x, recursive = FALSE)
```

Arguments

x	The object to test.
recursive	Unused.

Value

A flag indicating whether there are any missing values.

See Also

[base::anyNA](#)

Examples

```
anyNA(mcmcr_example)  
anyNA(mcmcr_example$beta)
```

as.mccarray	<i>Coerce to an mccarray object</i>
-------------	-------------------------------------

Description

Coerces MCMC objects to an `mccarray` object.

Usage

```
as.mccarray(x, ...)
```

```
## S3 method for class 'mcmcarray'
as.mccarray(x, ...)
```

Arguments

x	object to coerce.
...	Unused.

Methods (by class)

- `mcmcarray`: Coerces `mcmcarray` object to an `mccarray` object

Examples

```
as.mccarray(mcmcr_example$beta)
```

as.mcmc	<i>Coerce to an mcmc object</i>
---------	---------------------------------

Description

Coerces MCMC objects to an `mcmc` object.

Usage

```
## S3 method for class 'mccarray'
as.mcmc(x, ...)
```

```
## S3 method for class 'mcmcarray'
as.mcmc(x, ...)
```

```
## S3 method for class 'mcmc.list'
as.mcmc(x, ...)
```

```
## S3 method for class 'mcmcr'
as.mcmc(x, ...)
```

Arguments

x object to coerce.
 ... Unused.

Methods (by class)

- marray: Coerces marray object (with 1 chain) to an mcmc object
- mcmarray: Coerces mcmarray object (with 1 chain) to an mcmc object
- mcmc.list: Coerces mcmc.list object (with 1 chain) to an mcmc object
- mcmcr: Coerces mcmcr object (with 1 chain) to an mcmc object

See Also

coda::mcmc

Examples

```
as.mcmc(subset(mcmcr_example, chains = 1L))
```

as.mcmc.list	<i>Coerce to an mcmc.list object</i>
--------------	--------------------------------------

Description

Coerces MCMC objects to an [mcmc.list](#) object.

Usage

```
## S3 method for class 'marray'
as.mcmc.list(x, ...)

## S3 method for class 'mcmarray'
as.mcmc.list(x, ...)

## S3 method for class 'mcmc'
as.mcmc.list(x, ...)

## S3 method for class 'mcmcr'
as.mcmc.list(x, ...)
```

Arguments

x object to coerce.
 ... Unused.

Methods (by class)

- `mccarray`: Coerces `mccarray` object to an `mcmc.list` object
- `mcmcarray`: Coerces `mcmcarray` object to an `mcmc.list` object
- `mcmc`: Coerces `mcmc` object to an `mcmc.list` object
- `mcmcr`: Coerces `mcmcr` object to an `mcmc.list` object

`as.mcmcarray`
Coerce to an mcmcarray object

Description

Coerces MCMC objects to an [mcmcarray-object](#).

Usage

```
as.mcmcarray(x, ...)
```

Default S3 method:
`as.mcmcarray(x, ...)`

S3 method for class 'mccarray'
`as.mcmcarray(x, ...)`

S3 method for class 'mcmc'
`as.mcmcarray(x, ...)`

S3 method for class 'mcmc.list'
`as.mcmcarray(x, ...)`

S3 method for class 'mcmcr'
`as.mcmcarray(x, ...)`

Arguments

`x` object to coerce.
`...` Unused.

Methods (by class)

- `default`: Coerces vector, matrix or array to an `mcmcarray` object
- `mccarray`: Coerces `mccarray` object to an `mcmcarray` object
- `mcmc`: Coerces `mcmc` object (with one parameter) to an `mcmcarray` object
- `mcmc.list`: Coerces `mcmc.list` object (with one parameter) to an `mcmcarray` object
- `mcmcr`: Coerces `mcmcr` object (with one parameter) to an `mcmcarray` object

Examples

```
as.mcmcrarray(as.mccarray(mcmcr_example$beta))
```

as.mcmcr	<i>Coerce to an mcmcr object</i>
----------	----------------------------------

Description

Coerces MCMC objects to an `mcmcr-object`.

Usage

```
as.mcmcr(x, ...)

## S3 method for class 'list'
as.mcmcr(x, ...)

## S3 method for class 'mccarray'
as.mcmcr(x, ...)

## S3 method for class 'mcmc'
as.mcmcr(x, ...)

## S3 method for class 'mcmc.list'
as.mcmcr(x, ...)

## S3 method for class 'mcmccarray'
as.mcmcr(x, ...)

## S3 method for class 'list'
as.mcmcrs(x, ...)
```

Arguments

x	object to coerce.
...	Unused.

Methods (by class)

- `list`: Coerces list (of `mcmccarray` objects) to an `mcmcr` object
- `mccarray`: Coerces `mccarray` object to an `mcmcr` object
- `mcmc`: Coerces `mcmc` object to an `mcmcr` object
- `mcmc.list`: Coerces `mcmc.list` object to an `mcmcr` object
- `mcmccarray`: Coerces `mcmccarray` object to an `mcmcr` object
- `list`: Coerces list (of `mcmcr` objects with the same parameters, chains and iterations) to an `mcmcrs` object

Examples

```
as.mcmc(coda::as.mcmc.list(mcmc_example))
```

as.mcmcrs	<i>Coerce to an mcmcrs object</i>
-----------	-----------------------------------

Description

Coerces MCMC objects to an [mcmcrs-object](#).

Usage

```
as.mcmcrs(x, ...)
```

Arguments

x	object to coerce.
...	Unused.

Examples

```
as.mcmcrs(list(mcmc_example))
```

as.term	<i>Coerce to a term vector</i>
---------	--------------------------------

Description

Coerces MCMC objects to a term vector.

Usage

```
as.term(x, ...)

## S3 method for class 'term'
as.character(x, ...)

## S3 method for class 'character'
as.term(x, ...)

## S3 method for class 'mcmc'
as.term(x, ...)

## S3 method for class 'mcmc.list'
as.term(x, ...)
```



```
## S3 method for class 'mcmcarray'
as.term(x, ...)

## S3 method for class 'mcmcr'
as.term(x, ...)
```

Arguments

x	The object to coerce
...	Unused.

Methods (by class)

- term: Coerces term vector to a character vector
- character: Coerces character vector to a term vector
- mcmc: Coerces mcmc object to a term vector
- mcmc.list: Coerces mcmc.list object to a term vector
- mcmcarray: Coerces mcmcarray object to a term vector
- mcmcr: Coerces mcmcr object to a term vector

Examples

```
as.term(mcmcr_example)
```

bind_chains	<i>Combine MCMC objects by chains.</i>
-------------	--

Description

Combines two MCMC objects (with the same parameters and iterations) by chains.

Usage

```
bind_chains(x, x2, ...)

## S3 method for class 'mccarray'
bind_chains(x, x2, ...)

## S3 method for class 'mcmc'
bind_chains(x, x2, ...)

## S3 method for class 'mcmc.list'
bind_chains(x, x2, ...)

## S3 method for class 'mcmcarray'
```

```
bind_chains(x, x2, ...)

## S3 method for class 'mcmcr'
bind_chains(x, x2, ...)
```

Arguments

x	an MCMC object.
x2	a second MCMC object
...	Unused.

Methods (by class)

- marray: Binds two marray objects by their chains
- mcmc: Binds two mcmc objects by their chains
- mcmc.list: Binds two mcmc.list objects by their chains
- mcmarray: Binds two mcmarray objects by their chains
- mcmcr: Binds two mcmcr objects by their chains

Examples

```
bind_chains(mcmcr_example, mcmcr_example)
```

bind_dimensions	<i>Combine two MCMC objects by dimensions</i>
-----------------	---

Description

Combines multiple MCMC objects (with the same parameters, chains and iterations) by parameter dimensions.

Usage

```
bind_dimensions(x, x2, along = NULL, ...)

## S3 method for class 'mcmarray'
bind_dimensions(x, x2, along = NULL, ...)

## S3 method for class 'mcmcr'
bind_dimensions(x, x2, along = NULL, ...)
```

Arguments

x	An MCMC object.
x2	a second MCMC object.
along	A count (or NULL) indicating the parameter dimension to bind along.
...	Unused.

Methods (by class)

- `mcmcarray`: Binds two `mcmcarray` objects by their dimensions
- `mcmcr`: Binds two `mcmcr` objects by their dimensions

See Also

[bind_dimensions_n](#)

Examples

```
bind_dimensions(mcmcr_example, mcmcr_example)
```

<code>bind_dimensions_n</code>	<i>Combine multiple MCMC objects by parameter dimensions</i>
--------------------------------	--

Description

Combines multiple MCMC objects (with the same parameters, chains and iterations) by parameter dimensions.

Usage

```
bind_dimensions_n(...)

## S3 method for class 'mcmcarray'
bind_dimensions_n(...)

## S3 method for class 'mcmcr'
bind_dimensions_n(...)
```

Arguments

... one or more MCMC objects

Methods (by class)

- `mcmcarray`: Binds multiple `mcmcarray` objects by their dimensions
- `mcmcr`: Binds multiple `mcmcr` objects by their dimensions

See Also

[bind_dimensions](#)

Examples

```
bind_dimensions_n(mcmcr_example, mcmcr_example, mcmcr_example)
```

bind_iterations	<i>Combine two MCMC objects by iterations</i>
-----------------	---

Description

Combines two MCMC objects (with the same parameters and chains) by iterations.

Usage

```
bind_iterations(x, x2, ...)  
  
## S3 method for class 'mccarray'  
bind_iterations(x, x2, ...)  
  
## S3 method for class 'mcmc'  
bind_iterations(x, x2, ...)  
  
## S3 method for class 'mcmc.list'  
bind_iterations(x, x2, ...)  
  
## S3 method for class 'mcmcarray'  
bind_iterations(x, x2, ...)  
  
## S3 method for class 'mcmcr'  
bind_iterations(x, x2, ...)
```

Arguments

x	an MCMC object
x2	a second MCMC object
...	unused

Methods (by class)

- `mccarray`: Binds two `mccarray` objects by their iterations
- `mcmc`: Binds two `mcmc` objects by their iterations
- `mcmc.list`: Binds two `mcmc.list` objects by their iterations
- `mcmcarray`: Binds two `mcmcarray` objects by their iterations
- `mcmcr`: Binds two `mcmcr` objects by their iterations

Examples

```
bind_iterations(mcmcr_example, mcmcr_example)
```

bind_parameters	<i>Combine two MCMC object by parameters</i>
-----------------	--

Description

Combines two MCMC objects (with the same chains and iterations) by their parameters.

Usage

```
bind_parameters(x, ...)  
  
## S3 method for class 'mcmc'  
bind_parameters(x, x2, ...)  
  
## S3 method for class 'mcmc.list'  
bind_parameters(x, x2, ...)  
  
## S3 method for class 'mcmcr'  
bind_parameters(x, x2, ...)
```

Arguments

x	an MCMC object
...	unused
x2	a second MCMC object

Methods (by class)

- `mcmc`: Binds two `mcmc` objects by their parameters
- `mcmc.list`: Binds two `mcmc.list` objects by their parameters
- `mcmcr`: Binds two `mcmcr` objects by their parameters

Examples

```
bind_parameters(subset(mcmcr_example, parameters = "sigma"),  
               subset(mcmcr_example, parameters = "beta"))
```

coef *Term Coefficients*

Description

Gets coefficients for all the terms in an MCMC object.

Usage

```
## S3 method for class 'mccarray'
coef(object, conf_level = 0.95, estimate = stats::median,
     ...)

## S3 method for class 'mcmc'
coef(object, conf_level = 0.95, estimate = stats::median,
     ...)

## S3 method for class 'mcmc.list'
coef(object, conf_level = 0.95,
     estimate = stats::median, ...)

## S3 method for class 'mcmcarray'
coef(object, conf_level = 0.95,
     estimate = stats::median, ...)

## S3 method for class 'mcmcr'
coef(object, conf_level = 0.95, estimate = stats::median,
     ...)
```

Arguments

object	The MCMC object to get the coefficients for
conf_level	A number specifying the confidence level. By default 0.95.
estimate	The function to use to calculate the estimate.
...	Unused

Value

An tibble of the coefficients with the columns indicating the term, estimate, standard deviation (sd), zscore, lower and upper credible intervals and pvalue.

Methods (by class)

- `mccarray`: Get coefficients for terms in `mccarray` object
- `mcmc`: Get coefficients for terms in `mcmc` object
- `mcmc.list`: Get coefficients for terms in `mcmc.list` object

- `mcmcarray`: Get coefficients for terms in `mcmcarray` object
- `mcmcr`: Get coefficients for terms in `mcmcr` object

See Also

`stats::coef`

Examples

```
coef(mcmcr_example)
```

collapse_chains	<i>Collapse Chains</i>
-----------------	------------------------

Description

Collapses an MCMC object's chains into a single chain.

Usage

```
collapse_chains(x, ...)  
  
## Default S3 method:  
collapse_chains(x, ...)  
  
## S3 method for class 'mcmc.list'  
collapse_chains(x, ...)  
  
## S3 method for class 'mcmcr'  
collapse_chains(x, ...)
```

Arguments

<code>x</code>	An MCMC object.
<code>...</code>	Unused.

Methods (by class)

- `default`: Collapse an MCMC object's chains into a single chain
- `mcmc.list`: Collapse an `mcmc.list` object's chains into a single chain
- `mcmcr`: Collapse an `mcmcr` object's chains into a single chain

Examples

```
collapse_chains(mcmcr_example)
```

combine_dimensions *Combine Samples by Dimensions*

Description

Combines MCMC object samples by dimensions along along using fun.

Usage

```
combine_dimensions(x, fun = mean, along = NULL, ...)  
  
## S3 method for class 'mcmcarray'  
combine_dimensions(x, fun = mean, along = NULL, ...)  
  
## S3 method for class 'mcmcr'  
combine_dimensions(x, fun = mean, along = NULL, ...)
```

Arguments

x	An MCMC object
fun	The function to use when combining dimensions
along	A positive integer (or NULL) indicating the parameter dimension(s) to bind along.
...	Unused

Value

The MCMC object with reduced dimensions.

Methods (by class)

- mcmcarray: Combine an mcmcarray object's samples by dimensions
- mcmcr: Combine an mcmcr object's samples by dimensions

Examples

```
combine_dimensions(mcmcr_example$alpha)
```

combine_samples	<i>Combine MCMC Samples of Two Objects</i>
-----------------	--

Description

Combines samples of two MCMC objects (with the same parameters, chains and iterations) using a function.

Usage

```
combine_samples(x, x2, fun = mean, ...)  
  
## S3 method for class 'mcmcarray'  
combine_samples(x, x2, fun = mean, ...)  
  
## S3 method for class 'mcmcr'  
combine_samples(x, x2, fun = mean, ...)
```

Arguments

x	An MCMC object.
x2	A second MCMC object.
fun	The function to use to combine the samples. The function must return a scalar.
...	Unused.

Value

The combined samples as an MCMC object with the same parameters, chains and iterations as the original objects.

Methods (by class)

- mcmcarray: Combine samples of two mcmcarray objects
- mcmcr: Combine samples of two mcmcr objects

Examples

```
combine_samples(mcmcr_example, mcmcr_example, fun = sum)
```

combine_samples_n	<i>Combine MCMC Samples of multiple objects</i>
-------------------	---

Description

Combines samples of multiple MCMC objects (with the same parameters, chains and iterations) using a function.

Usage

```
combine_samples_n(x, ..., fun = mean)

## Default S3 method:
combine_samples_n(x, ..., fun = mean)

## S3 method for class 'list'
combine_samples_n(x, ..., fun = mean)
```

Arguments

x	An MCMC object (or a list of mcmc objects).
...	Additional MCMC objects.
fun	A function.

Methods (by class)

- default: Combine samples of multiple MCMC objects
- list: Combine samples of a list of multiple MCMC objects

Examples

```
combine_samples_n(mcmcr_example, mcmcr_example, mcmcr_example, fun = sum)
```

converged	<i>Object Converged</i>
-----------	-------------------------

Description

Test whether an object has converged.

Usage

```
converged(x, ...)

## Default S3 method:
converged(x, rhat = 1.1, esr = 0.33, by = "all",
  as_df = FALSE, ...)
```

Arguments

x	An object.
...	Unused.
rhat	The maximum rhat value.
esr	The minimum effective sampling rate.
by	A string indicating whether to determine by "term", "parameter" or "all".
as_df	A flag indicating whether to return the results as a (tbl) data frame.

Methods (by class)

- default: Test whether an object has converged

Examples

```
converged(mcmcr_example)
```

dims

Dimensions

Description

Dimensions

Usage

```
dims(x)
```

Arguments

x	A vector, matrix or array.
---	----------------------------

Value

An integer vector of the dimensions.

Examples

```
dims(character())  
dims(2:3)  
dims(matrix(1:6, nrow = 2))
```

 esr

Effective Sampling Rate

Description

Calculates the effective sampling rate (esr) based on the formula

$$\frac{1}{1 + 2 \sum_{k=1}^{\infty} \rho_k(\theta)}$$

in Brooks et al. (2011). The infinite sum is truncated at lag k when $\rho_{k+1}(\theta) < 0$.

Usage

```
esr(x, ...)

## S3 method for class 'marray'
esr(x, by = "all", ...)

## S3 method for class 'mcmc'
esr(x, by = "all", ...)

## S3 method for class 'mcmc.list'
esr(x, by = "all", ...)

## S3 method for class 'mcmcarray'
esr(x, by = "all", as_df = FALSE, ...)

## S3 method for class 'mcmcr'
esr(x, by = "all", as_df = FALSE, ...)

## S3 method for class 'mcmcrs'
esr(x, by = "all", ...)
```

Arguments

x	An MCMC object
...	Unused
by	A string indicating whether to return the estimates by the object ("all"), "parameter" or "term"
as_df	A flag indicating whether to return the estimates as a tibble versus a list.

Value

The esr value(s) as a tibble or list

Methods (by class)

- `mccarray`: Effective Sampling Rate for an `mccarray` object
- `mcmc`: Effective Sampling Rate for an `mcmc` object
- `mcmc.list`: Effective Sampling Rate for an `mcmc.list` object
- `mcmcarray`: Effective Sampling Rate for an `mcmcarray` object
- `mcmcr`: Effective Sampling Rate for an `mcmcr` object
- `mcmcrs`: Effective Sampling Rate for an `mcmcrs` object

References

Brooks, S., Gelman, A., Jones, G.L., and Meng, X.-L. (Editors). 2011. Handbook for Markov Chain Monte Carlo. Taylor & Francis, Boca Raton.

Examples

```
ess(mcmcr_example)
```

ess

Effective Sample Size

Description

Calculates the effective sample size based on [essr](#).

Usage

```
ess(x, by = "all", as_df = FALSE)
```

Arguments

<code>x</code>	An MCMC object.
<code>by</code>	A string indicating whether to determine by "term", "parameter" or "all".
<code>as_df</code>	A flag indicating whether to return the results as a tibble or list.

Examples

```
ess(mcmcr_example)
```

 estimates

Estimates

Description

Get the estimates for an MCMC object.

Usage

```
estimates(object, ...)

## S3 method for class 'mccarray'
estimates(object, fun = stats::median, as_df = FALSE, ...)

## S3 method for class 'mcmc'
estimates(object, fun = stats::median, as_df = FALSE, ...)

## S3 method for class 'mcmc.list'
estimates(object, fun = stats::median, as_df = FALSE,
  ...)

## S3 method for class 'mcmcarray'
estimates(object, fun = stats::median, as_df = FALSE,
  ...)

## S3 method for class 'mcmcr'
estimates(object, fun = stats::median, as_df = FALSE, ...)
```

Arguments

object	An MCMC object.
...	Unused.
fun	The function to use.
as_df	A flag indicating whether to return the estimates as a tibble versus a named list.

Value

A named list or tibble.

Methods (by class)

- `mccarray`: Estimates for an `mccarray` object
- `mcmc`: Estimates for an `mcmc` object
- `mcmc.list`: Estimates for an `mcmc.list` object
- `mcmcarray`: Estimates for an `mcmcarray` object
- `mcmcr`: Estimates for an `mcmcr` object

Examples

```
estimates(mcmcr_example)
estimates(mcmcr_example, as_df = TRUE)
```

is.marray	<i>Is marray Object</i>
-----------	-------------------------

Description

Tests whether an object is an marray.

Usage

```
is.marray(x)
```

Arguments

x The object to test.

Value

A flag indicating whether the test was positive.

Examples

```
is.marray(mcmcr_example)
```

is.mcmarray	<i>Is mcmarray Object</i>
-------------	---------------------------

Description

Tests whether an object is an [mcmarray-object](#).

Usage

```
is.mcmarray(x)
```

Arguments

x The object to test.

Value

A flag indicating whether the test was positive.

Examples

```
is.mcmarray(mcmcr_example$beta)
```

`is.mcmcr`*Is mcmcr Object*

Description

Tests whether an object is an [mcmcr-object](#).

Usage

```
is.mcmcr(x)
```

Arguments

`x` The object to test.

Value

A flag indicating whether the test was positive.

Examples

```
is.mcmcr(mcmcr_example)
```

`is.mcmcrs`*Is mcmcrs Object*

Description

Tests whether an object is an [mcmcrs-object](#).

Usage

```
is.mcmcrs(x)
```

Arguments

`x` The object to test.

Value

A flag indicating whether the test was positive.

Examples

```
is.mcmcrs(mcmcrs(mcmcr_example))
```

is.term	<i>Is Term</i>
---------	----------------

Description

Test whether an object is a term.

Usage

```
is.term(x)
```

Arguments

x The object to test.

Value

A flag indicating whether the test was positive.

Examples

```
is.term(terms(mcmcr_example))
```

mcmcarray-object	<i>mcmcarray</i>
------------------	------------------

Description

An mcmcarray object is an array where the first dimension is the chains, the second dimension is the iterations and the subsequent dimensions represent the dimensionality of the parameter. The name mcmcarray reflects the fact that the MCMC dimensions, ie the chains and iterations, precede the parameter dimensions.

Examples

```
mcmcr_example$beta
```

mcmcr-object	<i>mcmcr</i>
--------------	--------------

Description

An `mcmcr` object stores multiple uniquely named `mcmcrarray-object` objects with the same number of chains and iterations.

Details

`mcmcr` objects allow a set of dimensionality preserving parameters to be manipulated and queried as a whole.

Examples

```
mcmcr_example
```

mcmcrs	<i>Create mcmcrs</i>
--------	----------------------

Description

Creates an `mcmcrs-object` from multiple `link{mcmcr-object}s`.

Usage

```
mcmcrs(...)
```

Arguments

... Objects of class `mcmcr`.

Value

An object of class `mcmcrs`

Examples

```
mcmcrs(mcmcr_example, mcmcr_example)
```

`mcmcrs-object`*mcmcrs*

Description

An `mcmcrs` object stores multiple `mcmcr-object`s with the same parameters and the same number of chains and iterations.

Details

`mcmcrs` objects allow the results of multiple analyses using the same model to be manipulated and queried as a whole.

Examples

```
mcmcrs(mcmcr_example, mcmcr_example)
```

`mcmcr_example`*An Example mcmcr Object*

Description

An example `mcmcr-object` derived from on `coda::line`.

Usage

```
mcmcr_example
```

Format

An object of class `mcmcr` of length 3.

Examples

```
mcmcr_example
```

`mcmc_map`*MCMC Map*

Description

Adjust the sample values of an MCMC object using a function.

Usage

```
mcmc_map(.x, .f, ...)  
  
## Default S3 method:  
mcmc_map(.x, .f, ...)  
  
## S3 method for class 'mcmc.list'  
mcmc_map(.x, .f, ...)  
  
## S3 method for class 'mcmcr'  
mcmc_map(.x, .f, ...)  
  
## S3 method for class 'mcmcrs'  
mcmc_map(.x, .f, ...)
```

Arguments

<code>.x</code>	An MCMC object
<code>.f</code>	The function to use
<code>...</code>	Additional arguments passed to <code>.f</code> .

Value

The updated MCMC object.

Methods (by class)

- `default`: Adjust the sample values of an MCMC object
- `mcmc.list`: Adjust the sample values of an `mcmc.list` object
- `mcmcr`: Adjust the sample values of an `mcmcr` object
- `mcmcrs`: Adjust the sample values of an `mcmcrs` object

Examples

```
mcmc_map(mcmcr_example$beta, exp)
```

nchains	<i>Number of MCMC chains</i>
---------	------------------------------

Description

Gets the number of MCMC chains.

Usage

```
nchains(x, ...)  
  
## S3 method for class 'marray'  
nchains(x, ...)  
  
## S3 method for class 'mcmc'  
nchains(x, ...)  
  
## S3 method for class 'mcmc.list'  
nchains(x, ...)  
  
## S3 method for class 'mcmcarray'  
nchains(x, ...)  
  
## S3 method for class 'mcmcr'  
nchains(x, ...)  
  
## S3 method for class 'mcmcrs'  
nchains(x, ...)  
  
## S3 method for class 'marray'  
niters(x, ...)  
  
## S3 method for class 'mcmc'  
niters(x, ...)  
  
## S3 method for class 'mcmc.list'  
niters(x, ...)  
  
## S3 method for class 'mcmcarray'  
niters(x, ...)  
  
## S3 method for class 'mcmcr'  
niters(x, ...)  
  
## S3 method for class 'mcmcrs'  
niters(x, ...)
```

Arguments

x An MCMC object
 ... Unused

Value

A count indicating the number of MCMC chains

Methods (by class)

- `mccarray`: Number of MCMC chains for an `mccarray` object
- `mcmc`: Number of MCMC chains for an `mcmc` object
- `mcmc.list`: Number of MCMC chains for an `mcmc.list` object
- `mcmcarray`: Number of MCMC chains for an `mcmcarray` object
- `mcmcr`: Number of MCMC chains for an `mcmcr` object
- `mcmcrs`: Number of MCMC chains for an `mcmcrs` object
- `mccarray`: Number of MCMC iterations for an `mccarray` object
- `mcmc`: Number of MCMC iterations for an `mcmc` object
- `mcmc.list`: Number of MCMC iterations for an `mcmc.list` object
- `mcmcarray`: Number of MCMC iterations for an `mcmcarray` object
- `mcmcr`: Number of MCMC iterations for an `mcmcr` object
- `mcmcrs`: Number of MCMC iterations for an `mcmcrs` object

Examples

```
nchains(mcmcr_example)
```

<code>ndims</code>	<i>Number of dimensions</i>
--------------------	-----------------------------

Description

Number of dimensions

Usage

```
ndims(x)
```

Arguments

x A vector, matrix or array.

Value

A count of the number of dimensions

niters	<i>Number of MCMC samples</i>
--------	-------------------------------

Description

Gets the number of MCMC iterations (in a chain).

Usage

```
niters(x, ...)
```

Arguments

x	The object
...	Unused.

Value

A count indicating the number of MCMC iterations.

Examples

```
niters(mcmcr_example)
```

npars	<i>Number of Parameters</i>
-------	-----------------------------

Description

Gets the number of parameters for an object.

Usage

```
npars(x, ...)
```

```
## Default S3 method:  
npars(x, ...)
```

```
## S3 method for class 'mccarray'  
npars(x, ...)
```

```
## S3 method for class 'mcmcarray'  
npars(x, ...)
```

```
## S3 method for class 'mcmcr'  
npars(x, ...)
```

```
## S3 method for class 'mcmcrs'
npars(x, ...)
```

Arguments

x The object.
... Not used.

Value

A count of the number of parameters.

Methods (by class)

- default: Number of parameters for an object
- marray: Number of parameters for an marray object
- mcmarray: Number of parameters for an mcmarray object
- mcmcr: Number of parameters for an mcmcr object
- mcmcrs: Number of parameters for an mcmcrs object

Examples

```
npars(mcmcr_example)
```

nsams	<i>Number of MCMC Samples</i>
-------	-------------------------------

Description

The product of the number of simulations and the number of terms.

Usage

```
nsams(x)
```

Arguments

x The MCMC object.

Value

A count of the total number of samples.

Examples

```
nsams(mcmcr_example)
```

nsims	<i>Number of MCMC Simulations</i>
-------	-----------------------------------

Description

The product of the number of chains and number of iterations (in each chain).

Usage

```
nsims(x)
```

Arguments

x	The MCMC object.
---	------------------

Value

A count of the total number of simulations.

Examples

```
nsims(mcmcr_example)
```

nterms	<i>Number of Terms</i>
--------	------------------------

Description

Gets the number of terms for an object.

Usage

```
nterms(x, ...)
```

```
## Default S3 method:
```

```
nterms(x, ...)
```

```
## S3 method for class 'mcmcarray'
```

```
nterms(x, ...)
```

```
## S3 method for class 'mcmcr'
```

```
nterms(x, ...)
```

Arguments

x	The object.
...	Not used.

Value

A count of the number of terms.

Methods (by class)

- `default`: Number of terms for an object
- `mcmcarray`: Number of terms for an `mcmcarray` object
- `mcmcr`: Number of terms for an `mcmcr` object

Examples

```
nterms(mcmcr_example)
```

parameters

Parameter Names

Description

Gets or sets the parameter names for an object.

Usage

```
parameters(x, ...)
```

```
parameters(x) <- value
```

```
set_parameters(x, parameters)
```

```
## S3 method for class 'term'
parameters(x, scalar_only = FALSE, terms = FALSE, ...)
```

```
## S3 method for class 'mcmc'
parameters(x, scalar_only = FALSE, terms = FALSE, ...)
```

```
## S3 method for class 'mcmc.list'
parameters(x, scalar_only = FALSE, terms = FALSE, ...)
```

```
## S3 method for class 'mcmcr'
parameters(x, scalar_only = FALSE, terms = FALSE, ...)
```

Arguments

<code>x</code>	An MCMC object.
<code>...</code>	Not used.
<code>value</code>	A character vector of the new parameter names.
<code>parameters</code>	A character vector of the new parameter names.

scalar_only A flag indicating whether to only get the names of parameters with one term.
 terms A flag indicating whether to return the parameter name for each term.

Value

A character vector of the parameter names.

Methods (by class)

- term: Parameter names for a term vector
- mcmc: Parameter names for an mcmc object
- mcmc.list: Parameter names for an mcmc.list object
- mcmcr: Parameter names for an mcmcr object

Examples

```
parameters(mcmcr_example)
parameters(mcmcr_example) <- c("gamma", "theta", "tau")
parameters(mcmcr_example)
parameters(mcmcr_example, scalar_only = TRUE)
parameters(mcmcr_example, terms = TRUE)
```

 pdims

Parameter Dimensions

Description

Gets the parameter dimensions of an object.

Usage

```
pdims(x, ...)
```

S3 method for class 'term'

```
pdims(x, ...)
```

S3 method for class 'mccarray'

```
pdims(x, ...)
```

S3 method for class 'mcmc'

```
pdims(x, ...)
```

S3 method for class 'mcmc.list'

```
pdims(x, ...)
```

S3 method for class 'mcmcarray'

```
pdims(x, ...)
```

```
## S3 method for class 'mcmcr'
pdims(x, ...)
```

Arguments

x	The object
...	Unused.

Methods (by class)

- term: Parameter dimensions for a term vector
- marray: Parameter dimensions for an marray object
- mcmc: Parameter dimensions for an mcmc object
- mcmc.list: Parameter dimensions for an mcmc.list object
- mcmarray: Parameter dimensions for an mcmarray object
- mcmcr: Parameter dimensions for an mcmcr object

Examples

```
pdims(mcmcr_example)
```

pvalue	<i>P-Value</i>
--------	----------------

Description

Calculates the p-value.

Usage

```
pvalue(x)
```

Arguments

x	A vector of MCMC value
---	------------------------

Value

A number indicating the p-value.

Examples

```
pvalue(as.numeric(0:10))
```

rhat	<i>R-hat</i>
------	--------------

Description

Calculates the uncorrected, untransformed, univariate split R-hat (potential scale reduction factor) values.

Usage

```
rhat(x, ...)  
  
## S3 method for class 'marray'  
rhat(x, by = "all", as_df = FALSE, ...)  
  
## S3 method for class 'mcmc'  
rhat(x, by = "all", as_df = FALSE, ...)  
  
## S3 method for class 'mcmc.list'  
rhat(x, by = "all", as_df = FALSE, ...)  
  
## S3 method for class 'mcmcarray'  
rhat(x, by = "all", as_df = FALSE, ...)  
  
## S3 method for class 'mcmcr'  
rhat(x, by = "all", as_df = FALSE, ...)  
  
## S3 method for class 'mcmcrs'  
rhat(x, by = "all", as_df = FALSE, ...)
```

Arguments

x	An MCMC object.
...	Unused.
by	A string indicating whether to determine by "term", "parameter" or "all".
as_df	A flag indicating whether to return the values as a tibble versus a named list.

Value

The rhat value(s).

Methods (by class)

- `marray`: R-hat for an `marray` object
- `mcmc`: R-hat for an `mcmc` object
- `mcmc.list`: R-hat for an `mcmc.list` object

- `mcmcarray`: R-hat for an `mcmcarray` object
- `mcmcr`: R-hat for an `mcmcr` object
- `mcmcrs`: R-hat for an `mcmcrs` object

References

Gelman, A., and Rubin, D.B. 1992. Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science* 7(4): 457–472.

Examples

```
rhat(mcmcr_example)
rhat(mcmcr_example, by = "parameter")
rhat(mcmcr_example, by = "term")
rhat(mcmcr_example, by = "term", as_df = TRUE)
```

sort

Sort an MCMC Object

Description

Sorts an MCMC object by its parameter names.

Usage

```
## S3 method for class 'mcmc'
sort(x, ...)

## S3 method for class 'mcmc.list'
sort(x, ...)

## S3 method for class 'mcmcr'
sort(x, ...)

## S3 method for class 'mcmcrs'
sort(x, ...)
```

Arguments

<code>x</code>	The MCMC object to sort
<code>...</code>	Unused

Methods (by class)

- `mcmc`: Sort an `mcmc` object
- `mcmc.list`: Sort an `mcmc.list` object
- `mcmcr`: Sort an `mcmcr` object
- `mcmcrs`: Sort an `mcmcrs` object

Examples

```
parameters(mcmcr_example)
mcmcr_example <- subset(mcmcr_example, parameters = c("beta", "alpha"))
parameters(mcmcr_example)
mcmcr_example <- sort(mcmcr_example)
parameters(mcmcr_example)
```

split_chains

Split Chains

Description

Splits each chain in half to double the number chains and halve the number of iterations.

Usage

```
split_chains(x, ...)
```

S3 method for class 'mcmcarray'

```
split_chains(x, ...)
```

S3 method for class 'mcmcr'

```
split_chains(x, ...)
```

Arguments

x	An MCMC object.
...	Unused.

Methods (by class)

- mcmcarray: Split chains for an mcmcarray object
- mcmcr: Split chains for an mcmcr object

Examples

```
split_chains(mcmcr_example)
```

subset *Subset an MCMC Object*

Description

Subsets an MCMC object by its chains, iterations and/or parameters.

Usage

```
## S3 method for class 'term'
subset(x, parameters = NULL, ...)

## S3 method for class 'mcmc'
subset(x, iterations = NULL, parameters = NULL, ...)

## S3 method for class 'mcmc.list'
subset(x, chains = NULL, iterations = NULL,
       parameters = NULL, ...)

## S3 method for class 'mcmcarray'
subset(x, chains = NULL, iterations = NULL, ...)

## S3 method for class 'mcmcr'
subset(x, chains = NULL, iterations = NULL,
       parameters = NULL, ...)
```

Arguments

x	The MCMC object to subset
parameters	A character vector (or NULL) of the parameters to subset by.
...	Unused
iterations	An integer vector (or NULL) of the iterations to subset by.
chains	An integer vector (or NULL) of the chains to subset by.

Methods (by class)

- term: Subset a term vector
- mcmc: Subset an mcmc object
- mcmc.list: Subset an mcmc.list object
- mcmcarray: Subset an mcmcarray object
- mcmcr: Subset an mcmcr object

Examples

```
subset(mcmcr_example, chains = 2L, iterations = 1:100,
       parameters = c("beta", "alpha"))
```

tdims	<i>Term Dimensions</i>
-------	------------------------

Description

Gets the term dimensions of an object.

Usage

```
tdims(x, ...)  
  
## Default S3 method:  
tdims(x, ...)  
  
## S3 method for class 'term'  
tdims(x, ...)
```

Arguments

x	The object
...	Unused.

Methods (by class)

- default: Term dimensions for an object
- term: Term dimensions for a term vector

Examples

```
tdims(mcmcr_example)
```

term	<i>Term Vector</i>
------	--------------------

Description

A term vector is a sortable vector of the terms from an analysis.

Examples

```
terms <- terms(mcmcr_example)  
terms <- rev(terms)  
terms  
sort(terms)
```

terms

MCMC Object Terms

Description

Gets terms for an MCMC object.

Usage

```
## S3 method for class 'mcmc'  
terms(x, ...)  
  
## S3 method for class 'mcmc.list'  
terms(x, ...)  
  
## S3 method for class 'mcmcarray'  
terms(x, ...)  
  
## S3 method for class 'mcmcr'  
terms(x, ...)
```

Arguments

x	A MCMC object.
...	Unused

Value

A term vector.

Methods (by class)

- `mcmc`: Get terms for an `mcmc` object
- `mcmc.list`: Get terms for an `mcmc.list` object
- `mcmcarray`: Get terms for an `mcmcarray` object
- `mcmcr`: Get terms for an `mcmcr` object

See Also

`stats::terms`

Examples

```
terms(mcmcr_example)
```

thin	<i>Thin MCMC Samples</i>
------	--------------------------

Description

Thin an MCMC objects samples.

Usage

```
## S3 method for class 'mcmc'  
thin(x, nthin = 1L, ...)  
  
## S3 method for class 'mcmc.list'  
thin(x, nthin = 1L, ...)  
  
## S3 method for class 'mcmcarray'  
thin(x, nthin = 1L, ...)  
  
## S3 method for class 'mcmcr'  
thin(x, nthin = 1L, ...)  
  
## S3 method for class 'mcmcrs'  
thin(x, nthin = 1L, ...)
```

Arguments

x	An MCMC object
nthin	A count of the thinning rate.
...	Unused

Value

The thinned object.

Methods (by class)

- `mcmc`: Thin MCMC samples for an `mcmc` object
- `mcmc.list`: Thin MCMC samples for an `mcmc.list` object
- `mcmcarray`: Thin MCMC samples for an `mcmcarray` object
- `mcmcr`: Thin MCMC samples for an `mcmcr` object
- `mcmcrs`: Thin MCMC samples for an `mcmcrs` object

See Also

`coda::thin`

Examples

```
thin(mcmcr_example, nthin = 10L)
```

 zero

Zero MCMC Sample Values

Description

Zeros an MCMC object's sample values.

Usage

```
zero(x, ...)
```

```
## S3 method for class 'mccarray'
```

```
zero(x, ...)
```

```
## S3 method for class 'mcmcarray'
```

```
zero(x, ...)
```

```
## S3 method for class 'mcmcr'
```

```
zero(x, parameters = NULL, ...)
```

Arguments

x The MCMC object.

... Unused

parameters A character vector (or NULL) of the parameters to zero.

Details

It is used for removing the effect of a random effect where the expected value is 0.

Methods (by class)

- `mccarray`: Zero an `mccarray` object
- `mcmcarray`: Zero an `mcmcarray` object
- `mcmcr`: Zero an `mcmcr` object

Examples

```
zero(mcmcr_example, parameters = "beta")
```

Index

*Topic **datasets**

mcmcr_example, 27

anyNA, 3, 3

as.character.term (as.term), 8

as.marray, 4

as.mcmc, 4

as.mcmc.list, 5

as.mcmcarray, 6

as.mcmcr, 7

as.mcmcrs, 8

as.mcmcrs.list (as.mcmcr), 7

as.term, 8

bind_chains, 9

bind_dimensions, 10, 11

bind_dimensions_n, 11, 11

bind_iterations, 12

bind_parameters, 13

coef, 14, 15

collapse_chains, 15

combine_dimensions, 16

combine_samples, 17

combine_samples_n, 18

converged, 18

dims, 19

esr, 20, 21

ess, 21

estimates, 22

is.marray, 23

is.mcmcarray, 23

is.mcmcr, 24

is.mcmcrs, 24

is.term, 25

line, 27

marray.object, 4

mcmc, 4, 5

mcmc.list, 5

mcmc_map, 28

mcmcarray-object, 25

mcmcr-object, 26

mcmcr_example, 27

mcmcrs, 26

mcmcrs-object, 27

nchains, 29

ndims, 30

niters, 31

niters.marray (nchains), 29

niters.mcmc (nchains), 29

niters.mcmcarray (nchains), 29

niters.mcmcr (nchains), 29

niters.mcmcrs (nchains), 29

npars, 31

nsams, 32

nsims, 33

nterms, 33

parameters, 34

parameters<- (parameters), 34

pdims, 35

pvalue, 36

rhat, 37

set_parameters (parameters), 34

sort, 38

split_chains, 39

subset, 40

tdims, 41

term, 41

terms, 42, 42

thin, 43, 43

zero, 44