Package 'jagshelper'

November 8, 2022

Type Package

Title Extracting and Visualizing Output from 'jagsUI'

Version 0.1.11

Date 2022-11-08

Author Matt Tyers [aut, cre]

Maintainer Matt Tyers <matttyersstat@gmail.com>

Description Tools are provided to streamline Bayesian analyses in 'JAGS' using the 'jagsUI' package. Included are functions for extracting output in simpler format, functions for streamlining assessment of convergence, and functions for producing summary plots of output. Also included is a function that provides a simple template for running 'JAGS' from 'R'. Referenced materials can be found at <DOI:10.1214/ss/1177011136>.

License GPL-2

Encoding UTF-8

LazyData true

LazyDataCompression xz

RoxygenNote 7.2.1

Depends R (>= 3.5.0)

Imports jagsUI, MASS

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2022-11-08 15:40:02 UTC

R topics documented:

`jagshelper-package` *Functions for Extracting and Visualizing Output from 'jagsUI'*

Description

Functions are provided to help run Bayesian analyses in JAGS using the 'jagsUI' package. Included are functions for extracting output in simpler format, functions for streamlining assessment of convergence, and functions for producing summary plots of output. Also included is a function that provides a simple template for running JAGS from R.

Details

Package:	jagshelper
Type:	Package
Version:	0.1.11
Date:	2022-11-08
License:	GPL-2

The jagshelper package is intended to extend and streamline Bayesian analysis using the 'jagsUI' package.

The skeleton function prints a template JAGS model with associated R code to the console, which can easily be copied & pasted to an R script and modified as needed.

Functions are also provided for visually assessing model convergence. In particular, tracedens_jags gives a relatively simple syntax for trace plots of a collection or subset of parameter nodes, and overlays by-chain kernel densities for visual assessment of marginal posterior shapes as well as overlap between MCMC chains. Another function that could be particularly useful to users is plotRhats, which gives a visual representation of the values of the Gelman-Rubin convergence diagnostic `Rhat` (or alternately effective sample size `n.eff`) for all saved parameters. This may be particularly useful in the case where a model has many saved parameters. Additionally, function traceworstRhat is a wrapper for tracedens_jags, but only produces trace plots for the parameter nodes with the worst (largest) values of `Rhat` or `n.eff`.

Functions are also provided for visualizing posterior densities; in particular, the case of a vector of parameter nodes (one-dimensional in the JAGS model, giving a two-dimensional matrix of MCMC iterations). Notably, the envelope function is intended for a sequence of nodes (as in a time series), and the caterpillar function is intended for cases in which order may not matter (as in a collection of random effects).

Wrapper functions are also given for overlay of multiple such plots, as overlayenvelope and comparecat, and comparedens giving plots as vertically-oriented left- and right-facing kernel densities.

Author(s)

Matt Tyers

Maintainer: Matt Tyers <matttyersstat@gmail.com>

`asdf_jags_out` *Example data: asdf jags out*

Description

A simple model, equivalent to that produced by the output produced by `\link{skeleton}`.

Usage

```
asdf_jags_out
```

Format

An object of class jagsUI of length 24.

`caterpillar` *Caterpillar plot*

Description

Caterpillar plot of the posterior densities of a vector of parameter nodes, in which the sequential order of nodes might not be important, such as vector of random effects.

This produces a set of overlayed interval bars (default values are 50 percent and 95 percent), with overlayed median markings, for each of a vector of parameter nodes.

Usage

```
caterpillar(
  df,
  p = NULL,
  x = NA,
  row = NULL,
  column = NULL,
  median = TRUE,
  mean = FALSE,
  ci = c(0.5, 0.95),
  lwd = 1,
  col = 4,
  add = FALSE,
  xlab = "",
  ylab = "",
  main = NULL,
  xax = NA,
  medlwd = lwd,
  medwd = 1,
  ...
)
```

Arguments

df	Output object returned from jagsUI::jags(); or alternately, two-dimensional data.frame or matrix in which parameter node element is given by column and MCMC iteration is given by row.
p	Parameter name, if input to df is a jagsUI output object.
x	Vector of X-coordinates for plotting.
row	Row to subset, in the case of a 2-d matrix of parameter nodes in-model.
column	Column to subset, in the case of a 2-d matrix of parameter nodes in-model.

median	Whether to include medians
mean	Whether to include means
ci	Vector of intervals to overlay. Defaults to 50 percent and 95 percent.
lwd	Base line width for plotting. Defaults to 1.
col	Color for plotting
add	Whether to add to existing plot
xlab	X-axis label
ylab	Y-axis label
main	Plot title. If the default (NULL) is accepted and argument p is used, p will be used for the title.
xax	Vector of possible x-axis tick labels. Defaults to the data.frame column names.
medlwd	Line width of median line
medwd	Relative width of median line. Defaults to 1, perhaps smaller numbers will look better?
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

envelope

Examples

```
## usage with input data.frame
data(asdf_jags_out)
out_df <- jags_df(asdf_jags_out)

b1 <- pull_post(out_df,"b1")
a <- pull_post(out_df,"a")

caterpillar(a)
caterpillar(a, ci=seq(.1,.9,by=.1))
caterpillar(a, lwd=2)
caterpillar(a, xax=c("effect 1", "effect 2", "effect 3"))


## usage with input as jagsUI object
caterpillar(asdf_jags_out, p="a")
caterpillar(SS_out, p="rate")
```

```
## usage with a 2-d parameter matrix
caterpillar(SS_out, p="cycle_s", column=1)
caterpillar(SS_out, p="cycle_s", column=2)
```

`chaindens_df` *By-chain kernel density of each column of a* `data.frame`.

Description

By-chain kernel density plot of each column of a posterior `data.frame`.

Usage

```
chaindens_df(df, nline, parmfrow = NULL, ...)
```

Arguments

`df`	Posterior `data.frame`
`nline`	Number of chains
`parmfrow`	Optional call to `par(mfrow)` for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
`...`	additional plotting arguments or arguments

Value

`NULL`

Author(s)

Matt Tyers

See Also

tracedens_jags, trace_jags, trace_line

Examples

```
out_df <- jags_df(asdf_jags_out)

b1 <- pull_post(out_df,"b1")
a <- pull_post(out_df,"a")

trace_df(a, nline=3, parmfrow=c(3,1))

chaindens_df(a, nline=3, parmfrow=c(3,1))
```

chaindens_jags *By-chain kernel densities of* jagsUI *object*

Description

By-chain kernel densities of a whole jagsUI object, or optional subset of parameter nodes.

Usage

```
chaindens_jags(x, p = NULL, parmfrow = NULL, lwd = 1, ...)
```

Arguments

x	Posterior jagsUI object
p	Parameter name for subsetting: if this is specified, only parameters with names beginning with this string will be plotted.
parmfrow	Optional call to par(mfrow) for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
lwd	Line width for plotting. Defaults to 1.
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

tracedens_jags, trace_jags, chaindens_line, chaindens_df

Examples

```
chaindens_jags(asdf_jags_out, parmfrow=c(4,2))
chaindens_jags(x=asdf_jags_out, p="a", parmfrow=c(3,1))
```

chaindens_line *Simple by-chain kernel density plot*

Description

By-chain kernel density plot of a single parameter node.

Usage

```
chaindens_line(x, nline, lwd = 1, main = "", ...)
```

Arguments

x	Posterior vector
nline	Number of chains
lwd	Line width
main	Plot title
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

tracedens_jags, chaindens_jags, chaindens_df

Examples

```
out_df <- jags_df(asdf_jags_out)

b1 <- pull_post(out_df,"b1")
a <- pull_post(out_df,"a")

chaindens_line(b1, nline=3, main="b1")
```

check_neff *Quick summary of n.eff values by parameter name*

Description

Returns the mean number of `n.eff` values (by each parameter) that are greater than a specified threshold criterion.

`n.eff` is calculated within 'JAGS', and may be interpreted as a crude measure of effective sample size for a given parameter node.

Usage

```
check_neff(x, thresh = 500)
```

Arguments

`x`	Output object from `jagsUI::jags()`
`thresh`	Threshold value (defaults to 500)

Value

Numeric (named) giving the proportion of `n.eff` values above the given threshold.

Author(s)

Matt Tyers

See Also

check_Rhat

Examples

```
check_neff(SS_out)
```

check_Rhat *Quick summary of Rhat values by parameter name*

Description

Returns the mean number of `Rhat` values for each parameter (by each parameter) that are less than a specified threshold criterion.

`Rhat` (Gelman-Rubin Convergence Diagnostic, or Potential Scale Reduction Factor) is calculated within 'JAGS', and is commonly used as a measure of convergence for a given parameter node. Values close to 1 are seen as evidence of adequate convergence.

Usage

```
check_Rhat(x, thresh = 1.1)
```

Arguments

x	Output object from `jagsUI::jags()`
thresh	Threshold value (defaults to 1.1)

Value

Numeric (named) giving the proportion of Rhat values below the given threshold.

Author(s)

Matt Tyers

References

Gelman, A., & Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science, 7*(4), 457–472. http://www.jstor.org/stable/2246093

See Also

check_neff, traceworstRhat, plotRhats

Examples

```
check_Rhat(SS_out)
```

---

comparecat *Compare Caterpillar Plots*

Description

Interleaved caterpillar plots for all parameters (or a specified subset) from a list of `jagsUI` output objects or `data.frames`. The intent of this function is easy comparison of inferences from multiple comparable models.

Here a caterpillar plot is defined as a set of overlayed interval bars (default values are 50 percent and 95 percent), with overlayed median markings, for each of a vector of parameter nodes.

Usage

```
comparecat(x, p = NULL, ci = c(0.5, 0.95), ylim = NULL, ...)
```

Arguments

x	List of output objects returned from jagsUI or data.frames
p	Optional vector of parameters to subset. All parameters with names matching the beginning of the string supplied will be returned. If the default (NULL) is accepted, all parameters will be plotted.
ci	Credible intervals widths to plot. Defaults to 50% and 95%.
ylim	Y-axis limits for plotting
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

caterpillar, comparedens

Examples

```
## This is the same output object three times, but shows functionality.
comparecat(x=list(asdf_jags_out, asdf_jags_out, asdf_jags_out),
           p=c("a","b","sig"))
```

---

comparedens *Compare Density*

Description

Side-by-side kernel density plots for all parameters (or a specified subset) from two jagsUI output objects or data.frames. The intent of this function is easy comparison of inferences from two comparable models.

Kernel densities are plotted vertically, either left- or right-facing. Parameters with the same name are plotted facing one another.

Usage

```
comparedens(
  x1,
  x2,
  p = NULL,
  minCI = 0.99,
  legendnames = NULL,
```

```
  legendpos = "topleft",
  ...
)
```

Arguments

x1	Output object returned from jagsUI; or alternately a data.frame
x2	Second output object returned from jagsUI; or alternately a data.frame
p	Optional vector of parameters to subset. All parameters with names matching the beginning of the string supplied will be returned. If the default (NULL) is accepted, all parameters will be plotted.
minCI	Minimum CI width for plotting. This is intended as a method for excluding far-outlying MCMC samples when determining the appropriate y-axis limits for plotting. Defaults to 99%.
legendnames	Names for optional legend. If the default NULL is accepted, no legend will be drawn.
legendpos	Position for optional legend. Defaults to "topleft".
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

comparecat

Examples

```
## This is the same output object twice, but shows functionality.
comparedens(x1=asdf_jags_out, x2=asdf_jags_out, p=c("a","b","sig"),
            legendnames=c("Model 1", "Model 2"))
```

---

cor_jags *Correlation matrix from a JAGS object*

Description

Computes a correlation matrix of all MCMC samples from an object returned by 'jagsUI', or an optional subset of parameter nodes.

Usage

```
cor_jags(x, p = NULL, exact = FALSE)
```

Arguments

x	Output object returned from jagsUI
p	Optional string to begin posterior names. If NULL is used, all parameters will be used
exact	Whether name must be an exact match (TRUE) or with initial sub-string matching only supplied characters (FALSE). Defaults to FALSE.

Value

A 2-dimensional correlation matrix (n X n, where n is the number of parameter nodes)

Author(s)

Matt Tyers

See Also

plotcor_jags

Examples

```
cor_jags(asdf_jags_out)
```

---

envelope *Envelope plot*

Description

Envelope plot of the posterior densities of a vector of parameter nodes, in which the sequential order of nodes is important, such as a time series.

This produces a plot of overlayed shaded strips, each corresponding to a given interval width (defaults to 50 percent and 95 percent), with an overlayed median line.

Usage

```
envelope(
  df,
  p = NULL,
  x = NA,
  row = NULL,
  column = NULL,
  median = TRUE,
  ci = c(0.5, 0.95),
```

```
  col = 4,
  add = FALSE,
  dark = 0.3,
  outline = FALSE,
  xlab = "",
  ylab = "",
  main = NULL,
  ylim = NULL,
  ...
)
```

Arguments

df	Output object returned from `jagsUI::jags()`; or alternately, two-dimensional `data.frame` or matrix in which parameter node element is given by column and MCMC iteration is given by row.
p	Parameter name, if input to `df` is a `jagsUI` output object.
x	Vector of X-coordinates for plotting.
row	Row to subset, in the case of a 2-d matrix of parameter nodes in-model.
column	Column to subset, in the case of a 2-d matrix of parameter nodes in-model.
median	Whether to include median line
ci	Vector of intervals to overlay. Defaults to 50 percent and 95 percent.
col	Color for plotting
add	Whether to add to existing plot
dark	Opacity (0-1) for envelopes. Note that multiple overlapping intervals will darken the envelope.
outline	Whether to just envelope outlines
xlab	X-axis label
ylab	Y-axis label
main	Plot title. If the default (`NULL`) is accepted and argument p is used, p will be used for the title.
ylim	Y-axis limits for plotting. If the default (`NULL`) is accepted, these will be determined automatically.
...	additional plotting arguments or arguments to `lines()`

Value

`NULL`

Author(s)

Matt Tyers

See Also

overlayenvelope, caterpillar

Examples

```
## usage with input data.frame
SS_df <- jags_df(SS_out)
trend <- pull_post(SS_df, "trend")
envelope(trend, x=SS_data$x)

## usage with jagsUI object
envelope(SS_out, p="trend")

## usage with 2-d jagsUI object
envelope(SS_out, p="cycle_s", column=1, main="cycle")
envelope(SS_out, p="cycle_s", column=2, col=2, add=TRUE)  ## overlay
```

`expit` *Expit, or inverse logit*

Description

Inverse logit, where logit is defined as log(x/(1-x)).

Expit (inverse logit) is defined as exp(x)/(1+exp(x)).

Usage

```
expit(x)
```

Arguments

`x`	Numeric vector

Value

Numeric vector

Author(s)

Matt Tyers

See Also

logit

Examples

```
expit(0)
```

jags_df *Extract data.frame*

Description

Extracts the posterior samples from jagsUI output in the form of a data.frame. This simpler construction has a few benefits: operations may be more straightforward, and posterior objects will be smaller files and can be written to an external table or .csv, etc.

Usage

```
jags_df(x, p = NULL, exact = FALSE)
```

Arguments

x	Output object from jagsUI::jags()
p	Optional string to begin posterior names. If NULL is used, all parameters will be returned.
exact	Whether name must be an exact match (TRUE) or with initial sub-string matching only supplied characters (FALSE). Defaults to FALSE.

Value

A data.frame with a column associated with each parameter and a row associated with each MCMC iteration.

Author(s)

Matt Tyers

See Also

pull_post

Examples

```
out_df <- jags_df(asdf_jags_out)
```

jags_plist *Plist*

Description

Extracts a list of matrices, one for each saved parameter node. Each list element will be all posterior samples from that parameter node, arranged in a matrix with a column associated with each MCMC chain and a row for each MCMC iteration.

Usage

```
jags_plist(x, p = NULL)
```

Arguments

x	jagsUI output object
p	String to subset parameter names, if a subset is desired

Value

A list with an element associated with each parameter. Each element will be a matrix with a column associated with each MCMC chain and a row for each MCMC iteration.

Note

It is unlikely that a user will need this function; it is included primarily as a helper function used by other functions in this package.

Author(s)

Matt Tyers

Examples

```
out_plist <- jags_plist(asdf_jags_out)
str(out_plist)

a_plist <- jags_plist(asdf_jags_out, p=c("a","sig_a"))
str(a_plist)
```

logit *Logit*

Description

Logit log(x/(1-x))

Usage

```
logit(x)
```

Arguments

x	Numeric vector

Value

Numeric vector

Author(s)

Matt Tyers

See Also

expit

Examples

```
logit(0.5)
```

nbyname *Number of parameter nodes by parameter name*

Description

Returns a list of the numbers of parameter nodes saved in jagsUI output, by parameter name. As a default, what is returned for each list element is a vector of the array dimensions within the JAGS model (that is, excluding the dimension associated with the number of MCMC samples for each parameter node), or alternately, just the total number of parameter nodes.

Usage

```
nbyname(x, justtotal = FALSE)
```

Arguments

x	Output object from jagsUI::jags()
justtotal	Whether to just report the total number of parameters, as opposed to dimensions.

Value

A list with an element associated with each parameter. Each element can be interpreted as the vector length or array dimension associated with the given parameter.

Author(s)

Matt Tyers

See Also

nparam

Examples

```
head(jags_df(asdf_jags_out))

nbyname(asdf_jags_out)

nparam(SS_out)
nbyname(SS_out)
```

---

nparam *Number of parameters*

Description

Total number of individual parameter nodes saved in jagsUI output.

Usage

```
nparam(x)
```

Arguments

x	Output object from jagsUI::jags()

Value

A single numeric value giving the number of parameter nodes.

Author(s)

Matt Tyers

See Also

nbyname

Examples

```
head(jags_df(asdf_jags_out))

nparam(asdf_jags_out)
```

`overlayenvelope` *Overlay envelope plots*

Description

Overlays multiple envelope plots of posterior `data.frames`, or outputs returned from `jagsUI`. This would be best suited to a set of posterior `data.frames` or 2-d matrices representing sequential vectors of parameter nodes.

Here a single envelope plot is defined as a set of overlayed shaded strips, each corresponding to a given interval width (defaults to 50 percent and 95 percent), with an overlayed median line.

Usage

```
overlayenvelope(
  df,
  p = NULL,
  x = NA,
  row = NULL,
  column = NULL,
  median = TRUE,
  ci = c(0.5, 0.95),
  col = NULL,
  add = FALSE,
  dark = 0.3,
  outline = FALSE,
  xlab = "",
  ylab = "",
  main = NULL,
  ylim = NULL,
  legend = TRUE,
  legendnames = NULL,
  legendpos = "topleft",
  ...
)
```

Arguments

df	Primary input can be specified in a number of ways: either a `list()` of posterior `data.frames` or matrices, a `list` of output objects returned from `jagsUI::jags()`, a 3-dimensional `array` in which the input matrices to plot are separated according to the 3rd array dimension, or a single output object returned from `jagsUI::jags()` with multiple arguments passed to `p`, following.
p	Parameter name, if input to `df` is a list of `jagsUI` output objects; or a vector of parameter names, if input to `df` is a single `jagsUI` output object.
x	Optional vector of X-coordinates for plotting.
row	Row to subset, in the case of a 2-d matrix of parameter nodes in-model.
column	Column to subset, in the case of a 2-d matrix of parameter nodes in-model.
median	Whether to include median line
ci	Vector of intervals to overlay. Defaults to 50 percent and 95 percent.
col	Vector of colors for plotting
add	Whether to add to existing plot
dark	Opacity (0-1) for envelopes. Note that multiple overlapping intervals will darken the envelope. Defaults to 0.3.
outline	Whether to just envelope outlines
xlab	X-axis label
ylab	Y-axis label
main	Plot title. If the default (`NULL`) is accepted and argument `p=` is used, `p` will be used for the title.
ylim	Y-axis limits for plotting. If the default (`NULL`) is accepted, these will be determined automatically.
legend	Whether to automatically try to add a legend. Defaults to `TRUE`.
legendnames	Optional vector of names for a legend.
legendpos	Position for optional legend. Defaults to `"topleft"`.
...	additional plotting arguments or arguments to `lines()`

Value

`NULL`

Author(s)

Matt Tyers

See Also

envelope

Examples

```
## usage with list of input data.frames
overlayenvelope(df=list(SS_out$sims.list$cycle_s[,,1],
                        SS_out$sims.list$cycle_s[,,2]))

## usage with a 3-d input array
overlayenvelope(df=SS_out$sims.list$cycle_s)

## usage with a jagsUI output object and parameter name (2-d parameter)
overlayenvelope(df=SS_out, p="cycle_s")

## usage with a single jagsUI output object and multiple parameters
overlayenvelope(df=SS_out, p=c("trend","rate"))
```

---

pairstrace_jags *Pairs trace plot*

Description

Two-dimensional trace plots (or alternately, scatter plots or contour plots) of each possible pair of parameters from a possible subset. May be useful in assessing correlation between parameter nodes, or problematic posterior surfaces.

Usage

```
pairstrace_jags(
  x,
  p = NULL,
  points = FALSE,
  contour = FALSE,
  lwd = 1,
  alpha = 0.2,
  parmfrow = NULL,
  ...
)
```

Arguments

x	Output object returned from jagsUI
p	Optional vector of parameters to subset
points	Whether to plot as scatter plots instead. Defaults to FALSE.
contour	Whether to plot as contour plots instead. Defaults to FALSE.
lwd	Line width for trace plots. Defaults to 1.
alpha	Opacity of lines (or points, when points=TRUE). Defaults to 0.2.
parmfrow	Optional call to par(mfrow) for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
...	additional plotting arguments or arguments to tracedens_jags()

Value

NULL

Author(s)

Matt Tyers

See Also

tracedens_jags

Examples

```
pairstrace_jags(SS_out, p="sig", parmfrow=c(2,3), lwd=2)
pairstrace_jags(SS_out, p="sig", parmfrow=c(2,3), points=TRUE)
pairstrace_jags(SS_out, p="sig", parmfrow=c(2,3), contour=TRUE)

pairstrace_jags(asdf_jags_out, parmfrow=c(3,3))
pairstrace_jags(asdf_jags_out, parmfrow=c(3,3), points=TRUE)
pairstrace_jags(asdf_jags_out, parmfrow=c(3,3), contour=TRUE)
```

---

plotcor_jags *Plot a correlation matrix from a JAGS object*

Description

Plots a correlation matrix of all MCMC samples from an object returned by 'jagsUI', or an optional subset of parameter nodes. Correlation is plotted as shades of red (positive) or blue (negative).

In the case of vectors or arrays of nodes for each parameter name, a single axis tick will be used for all nodes with a single name. This has the effect of giving greater visual weight to single parameters, and reducing plot clutter.

Values of correlation are overlayed for all parameters with few nodes, with character size scaled according to the absolute correlation.

Usage

```
plotcor_jags(
  x,
  p = NULL,
  exact = FALSE,
  mincor = 0,
  maxn = 4,
  maxcex = 1,
  legend = TRUE,
  ...
)
```

Arguments

x	Output object returned from jagsUI
p	Optional string to begin posterior names. If NULL is used, all parameters will be used
exact	Whether name must be an exact match (TRUE) or with initial sub-string matching only supplied characters (FALSE). Defaults to FALSE.
mincor	Minimum (absolute) correlation to use for text labels. Defaults to 0 (all will be plotted)
maxn	Maximum number of nodes per parameter name for text labels, to prevent plot clutter. Defaults to 4.
maxcex	Maximum character expansion factor for text labels. Defaults to 1.
legend	Whether to produce a plot legend. Defaults to TRUE.
...	Optional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

plotcor_jags

Examples

```
plotcor_jags(asdf_jags_out, maxcex=0.7)

plotcor_jags(SS_out, p=c("trend","rate","sig"))
```

---

plotdens *Plot kernel densities of single parameter nodes*

Description

Produces a kernel density plot of a single or multiple parameter nodes (overlayed).

Input can be of multiple possible formats: either a single or list of output objects from jagsUI with an associated vector of parameter names, or a vector or data.frame of posterior samples.

Usage

```
plotdens(
  df,
  p = NULL,
  exact = FALSE,
  add = FALSE,
  col = NULL,
  shade = TRUE,
  lwd = 2,
  minCI = 0.99,
  legend = TRUE,
  legendpos = "topleft",
  legendnames = NULL,
  main = NULL,
  xlab = "",
  ylab = "Density",
  ...
)
```

Arguments

df	Input object for plotting. See examples below.
p	Vector of parameter names, if df is given as a single or list of output objects from jagsUI
exact	Whether the p= argument should match the parameter name exactly. See jags_df for details.
add	Whether to add to an existing plot (TRUE) or produce a new plot. Defaults to FALSE.
col	Vector of colors for plotting. If the default (NULL) is accepted, colors will be automatically selected.
shade	Whether to shade the regions below the kernel density curve(s). Defaults to TRUE.
lwd	Line width for kernel density curves. Defaults to 2. Note: setting this to 0 (or FALSE) will suppress lines.
minCI	Minimum CI width to include for all density curves. Defaults to 99%.
legend	Whether to plot a legend. Defaults to TRUE.
legendpos	Position for automatic legend. Defaults to "topleft".
legendnames	Names for legend
main	Plot title. Defaults to "".
xlab	X-axis label. Defaults to "".
ylab	Y-axis label. Defaults to "Density".
...	Optional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

comparedens, comparecat

Examples

```
## jagsUI object with a single parameter
plotdens(asdf_jags_out, p="b1")

## jagsUI object with multiple nodes of a parameter
plotdens(asdf_jags_out, p="a")

## jagsUI object with multiple parameter nodes
plotdens(asdf_jags_out, p=c("a[1]","a[2]","a[3]"))

## data.frame with multiple columns
plotdens(jags_df(asdf_jags_out, p="a"))

## list of jagsUI objects with a single parameter name
plotdens(list(asdf_jags_out,asdf_jags_out,asdf_jags_out), p="b1")

## list of jagsUI objects with a vector of parameter names
plotdens(list(asdf_jags_out,asdf_jags_out,asdf_jags_out), p=c("a[1]","a[2]","a[3]"))
```

---

plotRhats *Plotting all Rhat values*

Description

Plotting all values of Rhat (or alternately n.eff) from an output object returned by jagsUI, or perhaps a subset of parameters. This function is intended as a quick graphical check of which parameters have adequately converged.

Rhat (Gelman-Rubin Convergence Diagnostic, or Potential Scale Reduction Factor) is calculated within 'JAGS', and is commonly used as a measure of convergence for a given parameter node. Values close to 1 are seen as evidence of adequate convergence. n.eff is also calculated within 'JAGS', and may be interpreted as a crude measure of effective sample size for a given parameter node.

Usage

```
plotRhats(
  x,
  p = NULL,
  n.eff = FALSE,
  fence = NULL,
  plotsequence = FALSE,
  splitarr = FALSE,
  margin = NULL,
  ...
)
```

Arguments

x	Output object returned from jagsUI
p	Optional vector of parameters to subset
n.eff	Optionally, whether to plot n.eff instead of Rhat. Defaults to FALSE.
fence	Value of horizontal lines to overlay as reference. Accepting the default value (NULL) will give fence values of 1.1 (a commonly used value) and 1.01 for Rhat, or 100 and 500 for n.eff.
plotsequence	Whether to plot parameter vectors (or matrices) in a sequence, running left to right, which may be useful for time series models, etc. If the default (FALSE) is used, a vertical cluster will be plotted for each parameter, resulting in a simpler plot if there are many parameters. Note that the Rhat values will still be plotted in sequence if the default (FALSE) is used.
splitarr	Whether to split 2+ dimensional parameter arrays by a given dimension, rather than plotting the full array in one vertical cluster or continuous sequence. This may be recommended in the case of large arrays. Defaults to FALSE.
margin	If splitarr= is set to TRUE, which array margin to split by. In the case of a 2-dimensional array, setting margin=2 will separate the array by column. If the default (NULL) is accepted, the function will split by the smallest dimension, therefore splitting into the fewest groups.
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

References

Gelman, A., & Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science, 7*(4), 457–472. http://www.jstor.org/stable/2246093

See Also

traceworstRhat, check_Rhat

Examples

```
## plotting everything
plotRhats(SS_out)
str(SS_out$Rhat)  # the associated values

plotRhats(SS_out, n.eff=TRUE)
str(SS_out$n.eff)  # the associated values

## behavior of splitarr and margin are shown
plotRhats(SS_out)
plotRhats(SS_out, splitarr=TRUE)
str(SS_out$Rhat) # the associated values

## plotsequence may be useful in the case of a sequence of values
plotRhats(SS_out, p=c("trend", "cycle_s"), splitarr=TRUE, plotsequence=TRUE)
```

---

pull_post *Subset from posterior data.frame*

Description

Extracts a subset vector or `data.frame` from a `data.frame` consisting of more columns, such that column names match a name given in the `p=` argument. This may be useful in creating smaller objects consisting of MCMC samples.

Usage

```
pull_post(x, p = NULL, exact = FALSE)
```

Arguments

`x`	Posterior `data.frame`
`p`	String to begin posterior names. If `NULL` is used, all parameters will be returned.
`exact`	Whether name must be an exact match (`TRUE`) or with initial sub-string matching only supplied characters (`FALSE`). Defaults to `FALSE`.

Value

A `data.frame` with a column associated with each (subsetted) parameter and a row associated with each MCMC iteration.

Author(s)

Matt Tyers

See Also

jags_df

Examples

```
out_df <- jags_df(asdf_jags_out)

b <- pull_post(out_df, p="b")
str(b)
a <- pull_post(out_df, p=c("a","sig_a"))
str(a)
sigs <- pull_post(out_df, p="sig")
str(sigs)
justsig <- pull_post(out_df, p="sig", exact=TRUE)
str(justsig)
```

---

rcolors *Random Colors*

Description

Creates a vector of randomly-generated colors.

Usage

```
rcolors(n)
```

Arguments

n	Vector length

Value

A vector of colors

Author(s)

Matt Tyers

Examples

```
n <- 1000
cols <- rcolors(n)
x <- runif(n)
y <- runif(n)
plot(x,y, col=cols, pch=16)
```

skeleton *Skeleton*

Description

Prints an example 'JAGS' model and associated 'jagsUI' code to the console, along with code to simulate a corresponding dataset. This is intended to serve as a template that can be altered as needed by the user.

Usage

```
skeleton(NAME = "NAME")
```

Arguments

NAME	Name to append to JAGS model object, etc.

Value

NULL

Note

The printed code will use the `cat()` function to write the model code to an external text file. It may be desirable to use a call to `\link{tempfile}()` instead, to eliminate creation of unneeded files.

Author(s)

Matt Tyers

Examples

```
skeleton("asdf")
```

SS_data *Example data: Time series associated with SS JAGS out*

Description

The time series and time measurements associated with the time series model `\link{SS_out}`.

Usage

```
SS_data
```

Format

An object of class `data.frame` with 41 rows and 2 columns.

SS_out — *Example data: SS JAGS out*

Description

A time series model with multiple observations of a single time series, and with two stochastic cycle components.

Usage

```
SS_out
```

Format

An object of class `jagsUI` of length 24.

Details

This model is included partly to show a model with vectors or 2-dimensional matrices of parameter nodes, and also to give an example of poor model convergence.

tracedens_jags — *Combination of trace plots and by-chain kernel densities of* `jagsUI` *object*

Description

Combination of trace plots and by-chain kernel densities of a whole `jagsUI` object, or optional subset of parameter nodes.

Usage

```
tracedens_jags(x, p = NULL, parmfrow = NULL, lwd = 1, shade = TRUE, ...)
```

Arguments

`x`	Posterior `jagsUI` object
`p`	Parameter name for subsetting: if this is specified, only parameters with names beginning with this string will be plotted.
`parmfrow`	Optional call to `par(mfrow)` for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
`lwd`	Line width for plotting. Defaults to 1.
`shade`	Whether to add semi-transparent shading to by-chain kernel densities. Defaults to `TRUE`.
`...`	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

trace_jags, chaindens_jags, pairstrace_jags

Examples

```
tracedens_jags(asdf_jags_out, parmfrow=c(4,2))
tracedens_jags(asdf_jags_out, p="a", parmfrow=c(3,1))
```

---

traceworstRhat *Trace plots corresponding to the worst values of Rhat*

Description

Trace plots with kernel densities will be created for parameters with the largest (worst) associated values of Rhat. This function is primarily intended for parameters with a vector (or array) of values.

Rhat (Gelman-Rubin Convergence Diagnostic, or Potential Scale Reduction Factor) is calculated within 'JAGS', and is commonly used as a measure of convergence for a given parameter node. Values close to 1 are seen as evidence of adequate convergence. n.eff is also calculated within 'JAGS', and may be interpreted as a crude measure of effective sample size for a given parameter node.

Usage

```
traceworstRhat(x, p = NULL, n.eff = FALSE, margin = NULL, parmfrow = NULL, ...)
```

Arguments

x	Output object returned from jagsUI
p	Optional vector of parameters to subset
n.eff	Whether to plot parameters with the smallest associated values of n.eff instead. Defaults to FALSE.
margin	In the case of a 2+ dimensional array associated with a given parameter, this will have the effect of plotting the worst Rhat corresponding to each margin specified. For example, specifying margin=2 (column) will plot the parameter with the worst Rhat value from each column. In contrast, specifying margin=NULL (the default) will cause the function to plot the single array element with the largest Rhat value.

parmfrow	Optional call to par(mfrow) for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
...	additional plotting arguments or arguments to tracedens_jags()

Value

NULL

Author(s)

Matt Tyers

References

Gelman, A., & Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science, 7*(4), 457–472. http://www.jstor.org/stable/2246093

See Also

plotRhats, check_Rhat

Examples

```
## plotting everything
traceworstRhat(SS_out, parmfrow=c(3,2))
SS_out$Rhat  # the associated values

traceworstRhat(SS_out, parmfrow=c(3,2), n.eff=TRUE)
SS_out$n.eff  # the associated values

## in the case of a 2-D array, setting margin=2 gives the max Rhat
## associated with each column, rather than the global max
traceworstRhat(x=SS_out, p="cycle_s", margin=2, parmfrow=c(2,2))
SS_out$Rhat
traceworstRhat(x=SS_out, p="cycle_s", margin=2, parmfrow=c(2,2), n.eff=TRUE)
SS_out$n.eff
```

---

trace_df *Trace plot of each column of a* data.frame.

Description

Trace plot of each column of a posterior 'data.frame'.

Usage

```
trace_df(df, nline, parmfrow = NULL, ...)
```

Arguments

df	Posterior data.frame
nline	Number of chains
parmfrow	Optional call to par(mfrow) for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
...	additional plotting arguments or arguments to trace_line()

Value

NULL

Author(s)

Matt Tyers

See Also

tracedens_jags, trace_jags, trace_line

Examples

```
out_df <- jags_df(asdf_jags_out)

b1 <- pull_post(out_df,"b1")
a <- pull_post(out_df,"a")

trace_df(a, nline=3, parmfrow=c(3,1))

trace_df(a, nline=3, parmfrow=c(3,1))
```

---

trace_jags *Trace plot of jagsUI object*

Description

Trace plot of a whole jagsUI object, or optional subset of parameter nodes.

Usage

```
trace_jags(x, p = NULL, parmfrow = NULL, lwd = 1, ...)
```

Arguments

x	Posterior jagsUI object
p	Parameter name for subsetting: if this is specified, only parameters with names beginning with this string will be plotted.
parmfrow	Optional call to par(mfrow) for the number of rows & columns of plot window. Returns the graphics device to previous state afterward.
lwd	Line width for plotting. Defaults to 1.
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

tracedens_jags, pairstrace_jags, trace_df, trace_line

Examples

```
trace_jags(asdf_jags_out, parmfrow=c(4,2))
trace_jags(asdf_jags_out, p="a", parmfrow=c(3,1))
```

---

trace_line *Simple trace plot*

Description

Trace plot of a single parameter node.

Usage

```
trace_line(x, nline, lwd = 1, main = "", ...)
```

Arguments

x	Posterior vector
nline	Number of MCMC chains
lwd	Line width
main	Plot title
...	additional plotting arguments

Value

NULL

Author(s)

Matt Tyers

See Also

tracedens_jags, trace_jags, trace_df, chaindens_line

Examples

```
out_df <- jags_df(asdf_jags_out)

b1 <- pull_post(out_df,"b1")
a <- pull_post(out_df,"a")

trace_line(b1, nline=3, main="b1")
```

Index