Package ‘tracerer’

September 27, 2023

Type  Package
Title  Tracer from R
Version  2.2.3
Maintainer Richèl J.C. Bilderbeek <richel@richelbilderbeek.nl>
Description  'BEAST2' (<https://www.beast2.org>) is a widely used Bayesian phylogenetic tool, that uses DNA/RNA/protein data and many model priors to create a posterior of jointly estimated phylogenies and parameters. 'Tracer' (<https://github.com/beast-dev/tracer/>) is a GUI tool to parse and analyze the files generated by 'BEAST2'. This package provides a way to parse and analyze 'BEAST2' input files without active user input, but using R function calls instead.
License  GPL-3
Imports  jsonlite, Rcpp, testit
Suggests  ape, ggplot2, hunspell, knitr, markdown, phangorn, rappdirs, rbenchmark, reshape2, rmarkdown, spelling, stringr, testthat (>= 2.1.0)
VignetteBuilder  knitr
RoxygenNote  7.2.3
URL  https://docs.ropensci.org/tracerer/ (website)
    https://github.com/ropensci/tracerer/
BugReports  https://github.com/ropensci/tracerer/issues
LinkingTo  Rcpp
Language  en-US
Encoding  UTF-8
NeedsCompilation  yes
Author  Richèl J.C. Bilderbeek [aut, cre]
        (<https://orcid.org/0000-0003-1107-7049>),
        Joëlle Barido-Sottani [rev] (Joëlle reviewed the package for rOpenSci,
R topics documented:

see https://github.com/ropensci/onboarding/issues/209),
David Winter [rev] (David reviewed the package for rOpenSci, see
https://github.com/ropensci/onboarding/issues/209)

Repository CRAN

Date/Publication 2023-09-27 11:30:02 UTC

R topics documented:

calc_act .................................................. 3
 calc_act_cpp ......................................... 4
 calc_act_r ............................................ 4
 calc_ess ............................................. 5
 calc_esses ......................................... 6
 calc_geom_mean ...................................... 7
 calc_hpd_interval ................................... 7
 calc_mode ........................................... 8
 calc_stderr_mean ................................... 9
 calc_std_error_of_mean_cpp ....................... 10
 calc_summary_stats ................................. 10
 calc_summary_stats_trace ......................... 12
 calc_summary_stats_traces ....................... 13
 check_trace ......................................... 14
 count_trees_in_file ............................... 14
 cs_std_dev ......................................... 15
 default_params_doc .............................. 15
 extract_operators_lines ......................... 16
 get_tracerer_path ................................ 17
 get_tracerer_paths ................................ 18
 get_tracerer_tempfilename ....................... 19
 is_posterior ....................................... 19
 is_trees_file ...................................... 20
 is_trees_posterior ................................ 21
 parse_beast_log .................................. 21
 parse_beast_output_files ....................... 22
 parse_beast_posterior ......................... 23
 parse_beast_state_operators ................... 24
 parse_beast_tracelog_file ...................... 25
 parse_beast_trees ................................ 26
 remove_burn_in ................................... 26
 remove_burn_ins .................................. 27
 save_beast_estimates ......................... 28
 save_beast_trees ................................ 28

Index 30
calc_act

Calculate the auto-correlation time, alternative implementation

Description

Calculate the auto-correlation time, alternative implementation

Usage

calc_act(trace, sample_interval)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>trace</td>
<td>the values</td>
</tr>
<tr>
<td>sample_interval</td>
<td>the interval in timesteps between samples</td>
</tr>
</tbody>
</table>

Value

the auto_correlation time

Author(s)

The original Java version of the algorithm was from Remco Bouckaert, ported to R and adapted by Richèl J.C. Bilderbeek

See Also

Java code can be found here: [https://github.com/CompEvol/beast2/blob/9f040ed0357c4b946ea276a481a4c654ad4fa6/src/beast/core/util/ESS.java#L161](https://github.com/CompEvol/beast2/blob/9f040ed0357c4b946ea276a481a4c654ad4fa6/src/beast/core/util/ESS.java#L161) # nolint

Examples

```r
trace <- sin(seq(from = 0.0, to = 2.0 * pi, length.out = 100))
# 38.18202
calc_act(trace = trace, sample_interval = 1)
```
### Description

Calculate the auto correlation time from [https://github.com/beast-dev/beast-mcmc/blob/800817772033c13061f026226e41128d21fd14f3/src/dr/inference/trace/TraceCorrelation.java#L159](https://github.com/beast-dev/beast-mcmc/blob/800817772033c13061f026226e41128d21fd14f3/src/dr/inference/trace/TraceCorrelation.java#L159 # nolint)

### Usage

```r
calc_act_cpp(sample, sample_interval)
```

### Arguments

- **sample**: sample
- **sample_interval**: sample interval

### Value

the auto correlation time

### Author(s)

Richèl J.C. Bilderbeek

---

### Description

Calculate the auto-correlation time using only R. Consider using `calc_act` instead, as it is orders of magnitude faster

### Usage

```r
calc_act_r(trace, sample_interval)
```
Arguments

- `trace`: the values
- `sample_interval`: the interval in timesteps between samples

Value

the auto correlation time

Author(s)

The original Java version of the algorithm was from Remco Bouckaert, ported to R and adapted by Richèl J.C. Bilderbeek

See Also

Java code can be found here: https://github.com/CompEvol/beast2/blob/9f040ed0357c4b946ea276a481a4c654ad4f src/beast/core/util/ESS.java#L161 # nolint URLs can be long

Examples

```r
trace <- sin(seq(from = 0.0, to = 2.0 * pi, length.out = 100))
calc_act_r(trace = trace, sample_interval = 1) # 38.18202
```

Description

Calculates the Effective Sample Size

Usage

`calc_ess(trace, sample_interval)`

Arguments

- `trace`: the values without burn-in
- `sample_interval`: the interval in timesteps between samples

Value

the effective sample size

Author(s)

The original Java version of the algorithm was from Remco Bouckaert, ported to R and adapted by Richèl J.C. Bilderbeek
See Also

Java code can be found here: https://github.com/CompEvol/beast2/blob/9f040ed0357c4b946ea276a481a4c654ad4f src/beast/core/util/ESS.java#L161 # nolint URLs can be long

Examples

```r
filename <- get_tracerer_path("beast2_example_output.log")
estimates <- parse_beast_tracelog_file(filename)
calc_ess(estimates$posterior, sample_interval = 1000)
```

__calc_esses__

*Calculates the Effective Sample Sizes from a parsed BEAST2 log file*

Description

Calculates the Effective Sample Sizes from a parsed BEAST2 log file

Usage

calc_esses(traces, sample_interval)

Arguments

- **traces**: a dataframe with traces with removed burn-in
- **sample_interval**: the interval in timesteps between samples

Value

the effective sample sizes

Author(s)

Richèl J.C. Bilderbeek

Examples

```
# Parse an example log file
estimates <- parse_beast_tracelog_file(
  get_tracerer_path("beast2_example_output.log")
)

# Calculate the effective sample sizes of all parameter estimates
calc_esses(estimates, sample_interval = 1000)
```
### calc_geom_mean

*Calculate the geometric mean*

**Description**

Calculate the geometric mean

**Usage**

```r
calc_geom_mean(values)
```

**Arguments**

- `values` a numeric vector of values

**Value**

returns the geometric mean if all values are at least zero, else returns NA

**Author(s)**

Richèl J.C. Bilderbeek

---

### calc_hpd_interval

*Calculate the Highest Probability Density of an MCMC trace that has its burn-in removed*

**Description**

Calculate the Highest Probability Density of an MCMC trace that has its burn-in removed

**Usage**

```r
calc_hpd_interval(trace, proportion = 0.95)
```

**Arguments**

- `trace` a numeric vector of parameter estimates obtained from an MCMC run. Must have its burn-in removed
- `proportion` the proportion of numbers within the interval. For example, use 0.95 for a 95 percentage interval

**Value**

a numeric vector, with at index 1 the lower boundary of the interval, and at index 2 the upper boundary of the interval
**Author(s)**

The original Java version of the algorithm was from J. Heled, ported to R and adapted by Richèl J.C. Bilderbeek

**See Also**

The function `remove_burn_in` removes a burn-in. The Java code that inspired this function can be found here: [https://github.com/beast-dev/beast-mcmc/blob/98705c59db65e4f406a420bbade949aeecfe05d0/src/dr/stats/DiscreteStatistics.java#L317](https://github.com/beast-dev/beast-mcmc/blob/98705c59db65e4f406a420bbade949aeecfe05d0/src/dr/stats/DiscreteStatistics.java#L317) #nolint URLs can be long

**Examples**

```r
estimates <- parse_beast_tracelog_file(
  get_tracerer_path("beast2_example_output.log")
)
tree_height_trace <- remove_burn_in(
  estimates$TreeHeight,
  burn_in_fraction = 0.1
)

# Values will be 0.453 and 1.816
calc_hpd_interval(tree_height_trace, proportion = 0.95)
```

---

**calc_mode**  
*Calculate the mode of values If the distribution is bi or multimodal or uniform, NA is returned*

**Description**

Calculate the mode of values If the distribution is bi or multimodal or uniform, NA is returned

**Usage**

`calc_mode(values)`

**Arguments**

- `values` numeric vector to calculate the mode of

**Value**

the mode of the trace

**Author(s)**

Richèl J.C. Bilderbeek
calc_stderr_mean

Examples

# In a unimodal distribution, find the value that occurs most
calc_mode(c(1, 2, 2))
calc_mode(c(1, 1, 2))

# For a uniform distribution, NA is returned
tracerer:::calc_mode(c(1, 2))

calc_stderr_mean  Calculate the standard error of the mean

Description

Calculate the standard error of the mean

Usage

calc_stderr_mean(trace)

Arguments

trace      the values

Value

the standard error of the mean

Author(s)

The original Java version of the algorithm was from Remco Bouckaert, ported to R and adapted by
Richèl J.C. Bilderbeek

See Also

Java code can be found here: https://github.com/beast-dev/beast-mcmc/blob/800817772033c13061f026226e41128/
src/dr/inference/trace/TraceCorrelation.java#L159 # nolint URLs can be long

Examples

trace <- sin(seq(from = 0.0, to = 2.0 * pi, length.out = 100))
calc_stderr_mean(trace) # 0.4347425
calc_std_error_of_mean_cpp

*Calculates the standard error of the mean*

**Description**

Calculates the standard error of the mean

**Usage**

calc_std_error_of_mean_cpp(sample)

**Arguments**

- **sample**: numeric vector of values

**Value**

the standard error of the mean

**Author(s)**

Richèl J.C. Bilderbeek

---

calc_summary_stats

*Calculates the Effective Sample Sizes of one estimated variable’s trace.*

**Description**

Calculates the Effective Sample Sizes of one estimated variable’s trace.

**Usage**

calc_summary_stats(traces, sample_interval)

**Arguments**

- **traces**: one or more traces, supplies as either, (1) a numeric vector or, (2) a data frame of numeric values.
- **sample_interval**: the interval (the number of state transitions between samples) of the MCMC run that produced the trace. Using a different sample_interval than the actually used sampling interval will result in bogus return values.
Value
the summary statistics of the traces. If one numeric vector is supplied, a list is returned with the
elements listed below. If the traces are supplied as a data frame, a data frame is returned with the
elements listed below as column names.
The elements are:

- mean: mean
- stderr_mean: standard error of the mean
- stdev: standard deviation
- variance: variance
- mode: mode
- geom_mean: geometric mean
- hpd_interval_low: lower bound of 95% highest posterior density
- hpd_interval_high: upper bound of 95% highest posterior density
- act: auto correlation time
- ess: effective sample size

Note
This function assumes the burn-in is removed. Use remove_burn_in (on a vector) or remove_burn_ins
(on a data frame) to remove the burn-in.

Author(s)
Richèle J.C. Bilderbeek

See Also
Use calc_summary_stats_trace to calculate the summary statistics of one trace (stored as a
numeric vector). Use calc_summary_stats_traces to calculate the summary statistics of more
traces (stored as a data frame).

Examples
estimates_all <- parse_beast_tracelog_file(
    get_tracerer_path("beast2_example_output.log")
) estimates <- remove_burn_ins(estimates_all, burn_in_fraction = 0.1)

# From a single variable's trace
calc_summary_stats(
    estimates$posterior,
    sample_interval = 1000
)

# From all variables' traces
calc_summary_stats_trace

Calculates the Effective Sample Sizes of one estimated variable’s trace.

Description
Calculates the Effective Sample Sizes of one estimated variable’s trace.

Usage
calc_summary_stats_trace(trace, sample_interval)

Arguments
- **trace**: a numeric vector of values. Assumes the burn-in is removed.
- **sample_interval**: the interval in timesteps between samples

Value
the effective sample sizes

Author(s)
Richèl J.C. Bilderbeek

See Also
Use `remove_burn_in` to remove the burn-in of a trace

Examples
estimates_all <- parse_beast_tracelog_file(
  get_tracerer_path("beast2_example_output.log")
)
estimates <- remove_burn_ins(estimates_all, burn_in_fraction = 0.1)
calc_summary_stats_trace(
  estimates$posterior,
  sample_interval = 1000
)
calc_summary_stats_traces

Calculates the Effective Sample Sizes of the traces of multiple estimated variables.

Description
Calculates the Effective Sample Sizes of the traces of multiple estimated variables.

Usage
calc_summary_stats_traces(traces, sample_interval)

Arguments
- `traces`: a data frame with traces of estimated parameters. Assumes the burn-ins are removed.
- `sample_interval`: the interval in timesteps between samples

Value
the effective sample sizes

Author(s)
Richèl J.C. Bilderbeek

See Also
Use `remove_burn_ins` to remove the burn-ins of all traces

Examples
```r
estimates_all <- parse_beast_tracelog_file(get_tracerer_path("beast2_example_output.log"))
estimates <- remove_burn_ins(estimates_all, burn_in_fraction = 0.1)
calc_summary_stats_traces(
estimates,
sample_interval = 1000
)
```
count_trees_in_file

Count the number of trees in a .trees file

Description
Count the number of trees in a .trees file

Usage
count_trees_in_file(trees_filename)

Arguments
  trees_filename  name of a BEAST2 posterior .trees file, as can be read using parse_beast_trees

Value
  the number of trees

Author(s)
  Richèl J.C. Bilderbeek

See Also
  if the .trees file is invalid, use is_trees_file with verbose = TRUE for the reason

check_trace

Check if the trace is a valid. Will stop if not

Description
Check if the trace is a valid. Will stop if not

Usage
check_trace(trace)

Arguments
  trace  the values

Author(s)
  Richèl J.C. Bilderbeek

Examples
  check_trace(seq(1, 2))
**cs_std_dev**

*Calculate the corrected sample standard deviation.*

**Description**

Calculate the corrected sample standard deviation.

**Usage**

```python
cs_std_dev(values)
```

**Arguments**

- values: numeric values

**Value**

the corrected sample standard deviation

**Author(s)**

Richèl J.C. Bilderbeek

---

**default_params_doc**

*Documentation of general function arguments. This function does nothing. It is intended to inherit function argument documentation.*

**Description**

Documentation of general function arguments. This function does nothing. It is intended to inherit function argument documentation.

**Usage**

```python
default_params_doc(
    log_filename,
    sample_interval,
    state_filename,
    trace,
    tracelog_filename,
    trees_filename,
    trees_filenames,
    verbose
)
```
Arguments

log_filename  deprecated name of the BEAST2 tracelog .log output file. Use tracelog_filename instead
sample_interval  the interval in timesteps between samples
state_filename  name of the BEAST2 state .xml.state output file
trace  the values
tracelog_filename  name of the BEAST2 tracelog .log output file, as can be read using parse_beast_tracelog_file
trees_filename  name of a BEAST2 posterior .trees file, as can be read using parse_beast_trees
trees_filenames  the names of one or more a BEAST2 posterior .trees file. Each .trees file can be read using parse_beast_trees
verbose  set to TRUE for more output

Note

This is an internal function, so it should be marked with @noRd. This is not done, as this will disallow all functions to find the documentation parameters

Author(s)

Richèl J.C. Bilderbeek

extract_operators_lines

Extract the JSON lines out of a .xml.state with the unparsed BEAST2 MCMC operator acceptances file with the operators

Description

Extract the JSON lines out of a .xml.state with the unparsed BEAST2 MCMC operator acceptances file with the operators

Usage

extract_operators_lines(filename)

Arguments

filename  name of the BEAST2 .xml.state output file

Value

the JSON lines of a .xml.state file with the unparsed BEAST2 MCMC operator acceptances
get_tracerer_path

Author(s)
Richèl J.C. Bilderbeek

Description
Get the full path of a file in the inst/extdata folder

Usage
get_tracerer_path(filename)

Arguments
filename the file’s name, without the path

Value
the full path to the filename

Author(s)
Richèl J.C. Bilderbeek

See Also
for more files, use get_tracerer_paths

Examples
get_tracerer_path("beast2_example_output.log")
get_tracerer_path("beast2_example_output.trees")
get_tracerer_path("beast2_example_output.xml")
get_tracerer_path("beast2_example_output.xml.state")
**get_tracerer_paths**

Get the full paths of files in the `inst/extdata` folder

**Description**

Get the full paths of files in the `inst/extdata` folder

**Usage**

```r
get_tracerer_paths(filenames)
```

**Arguments**

- `filenames` : the files' names, without the path

**Value**

the filenames' full paths

**Author(s)**

Richèl J.C. Bilderbeek

**See Also**

for one file, use `get_tracerer_path`

**Examples**

```r
get_tracerer_paths(
  c(
    "beast2_example_output.log",
    "beast2_example_output.trees",
    "beast2_example_output.xml",
    "beast2_example_output.xml.state"
  )
)
```
get_tracerer_tempfilename

Get a temporary filename

Description

Get a temporary filename, similar to `tempfile`, except that it always writes to a temporary folder named `tracerer`.

Usage

```r
get_tracerer_tempfilename(pattern = "file", fileext = "")
```

Arguments

- `pattern` a non-empty character vector giving the initial part of the name.
- `fileext` a non-empty character vector giving the file extension

Value

name for a temporary file

Note

this function is added to make sure no temporary cache files are left undeleted

is_posterior

Determines if the input is a BEAST2 posterior

Description

Determines if the input is a BEAST2 posterior

Usage

```r
is_posterior(x)
```

Arguments

- `x` the input

Value

TRUE if the input contains all information of a BEAST2 posterior. Returns FALSE otherwise.
is_trees_file

**Description**
Measure if a file a valid BEAST2 .trees file

**Usage**
```r
is_trees_file(trees_filename, verbose = FALSE)
```

**Arguments**
- `trees_filename` name of a BEAST2 posterior .trees file, as can be read using `parse_beast_trees`
- `verbose` set to TRUE for more output

**Value**
TRUE if `trees_filename` is a valid .trees file

**Author(s)**
Richèl J.C. Bilderbeek

**See Also**
Most of the work is done by `read.nexus`

**Examples**
```r
# TRUE
is_trees_file(get_tracerer_path("beast2_example_output.trees"))
is_trees_file(get_tracerer_path("unplottable_anthus_aco.trees"))
is_trees_file(get_tracerer_path("anthurus_2_4_a.trees"))
is_trees_file(get_tracerer_path("anthurus_2_4_b.trees"))
# FALSE
is_trees_file(get_tracerer_path("mcbette_issue_8.trees"))
```
is_trees_posterior

Determines if the input is a BEAST2 posterior, as parsed by parse_beast_trees

Description
Determines if the input is a BEAST2 posterior, as parsed by parse_beast_trees

Usage
is_trees_posterior(x)

Arguments
x  the input

Value
TRUE or FALSE

Author(s)
Richèl J.C. Bilderbeek

parse_beast_log

Deprecated function to parse a BEAST2 .log output file. Use parse_beast_tracelog_file instead

Description
Deprecated function to parse a BEAST2 .log output file. Use parse_beast_tracelog_file instead

Usage
parse_beast_log(tracelog_filename, filename = "deprecated")

Arguments
tracelog_filename
name of the BEAST2 tracelog .log output file, as can be read using parse_beast_tracelog_file
filename  deprecated name of the BEAST2 .log output file

Value
data frame with the parameter estimates
parse_beast_output_files

Parse all BEAST2 output files

Description
Parse all BEAST2 output files

Usage
parse_beast_output_files(log_filename, trees_filenames, state_filename)

Arguments
log_filename
deprecated name of the BEAST2 tracelog .log output file. Use tracelog_filename instead

trees_filenames
the names of one or more a BEAST2 posterior .trees file. Each .trees file can be read using parse_beast_trees

state_filename
name of the BEAST2 state .xml.state output file

Value
a list with the following elements:

itemestimates: parameter estimates item [alignment_id]_trees: the phylogenies in the BEAST2 posterior. [alignment_id] is the ID of the alignment.

itemoperators: the BEAST2 MCMC operator acceptances

Author(s)
Richèl J.C. Bilderbeek
parse_beast_posterior

See Also

Use remove_burn_ins to remove the burn-in from out$estimates

Examples

trees_filenames <- get_tracerer_path("beast2_example_output.trees")
log_filename <- get_tracerer_path("beast2_example_output.log")
state_filename <- get_tracerer_path("beast2_example_output.xml.state")
parse_beast_output_files(
  log_filename = log_filename,
  trees_filenames = trees_filenames,
  state_filename = state_filename
)

parse_beast_posterior  Pares BEAST2 output files to a posterior

Description

Parses BEAST2 output files to a posterior

Usage

parse_beast_posterior(
  trees_filenames,
  tracelog_filename,
  log_filename = "deprecated"
)

Arguments

trees_filenames
  the names of one or more a BEAST2 posterior .trees file. Each .trees file
  can be read using parse_beast_trees

tracelog_filename
  name of the BEAST2 tracelog .log output file, as can be read using parse_beast_tracelog_file

log_filename
  deprecated name of the BEAST2 tracelog .log output file. Use tracelog_filename
  instead

Value

a list with the following elements:

itemestimates: parameter estimates item [alignment_id]_trees: the phylogenies in the
BEAST2 posterior. [alignment_id] is the ID of the alignment.
parse_beast_state_operators

Parses a BEAST2 state .xml.state output file to get only the operators acceptances

Description

Parses a BEAST2 state .xml.state output file to get only the operators acceptances

Usage

parse_beast_state_operators(
    state_filename = get_tracerer_path("beast2_example_output.xml.state"),
    filename = "deprecated"
)

Arguments

state_filename  name of the BEAST2 state .xml.state output file
filename        deprecated name of the BEAST2 .xml.state output file, use state_filename instead

Value

data frame with all the operators’ success rates

Author(s)

Richèl J.C. Bilderbeek
parse_beast_tracelog_file

Examples

```r
parse_beast_state_operators(
    state_filename = get_tracerer_path("beast2_example_output.xml.state")
)
```

---

parse_beast_tracelog_file

*Parses a BEAST2 tracelog *.log output file*

Description

Parses a BEAST2 tracelog `.log` output file

Usage

```
parse_beast_tracelog_file(tracelog_filename)
```

Arguments

- `tracelog_filename`
  - name of the BEAST2 tracelog `.log` output file, as can be read using `parse_beast_tracelog_file`

Value

- data frame with the parameter estimates

Author(s)

Richèl J.C. Bilderbeek

See Also

Use `remove_burn_ins` to remove the burn-in from the returned parameter estimates. Use `save_beast_estimates` to save the estimates to a `.log` file.

Examples

```r
parse_beast_tracelog_file(
    tracelog_filename = get_tracerer_path("beast2_example_output.log")
)
```
**parse_beast_trees**

*Parses a BEAST2 .trees output file*

**Description**

Parses a BEAST2 .trees output file

**Usage**

```r
parse_beast_trees(filename)
```

**Arguments**

- `filename` name of the BEAST2 .trees output file

**Value**

the phylogenies in the posterior

**Author(s)**

Richèl J.C. Bilderbeek

**See Also**

Use `save_beast_trees` to save the phylogenies to a .trees file. Use `is_trees_file` with `verbose = TRUE` to find out why a file is invalid

**Examples**

```r
trees_filename <- get_tracerer_path("beast2_example_output.trees")
parse_beast_trees(trees_filename)
```

---

**remove_burn_in**

*Removed the burn-in from a trace*

**Description**

Removed the burn-in from a trace

**Usage**

```r
remove_burn_in(trace, burn_in_fraction)
```
**Arguments**

<table>
<thead>
<tr>
<th>argument</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>trace</code></td>
<td>the values</td>
</tr>
<tr>
<td><code>burn_in_fraction</code></td>
<td>the fraction that needs to be removed, must be <code>[0,1&gt;</code></td>
</tr>
</tbody>
</table>

**Value**

the values with the burn-in removed

**Author(s)**

Richèl J.C. Bilderbeek

**Examples**

```r
# Create a trace from one to and including ten
t <- seq(1, 10)

# Remove the first ten percent of its values,
# in this case removes the first value, which is one
w <- remove_burn_in(trace = t, burn_in_fraction = 0.1)
```

---

**Description**

Removed the burn-ins from a data frame

**Usage**

`remove_burn_in(traces, burn_in_fraction = 0.1)`

**Arguments**

<table>
<thead>
<tr>
<th>argument</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>traces</code></td>
<td>a data frame with traces</td>
</tr>
<tr>
<td><code>burn_in_fraction</code></td>
<td>the fraction that needs to be removed, must be <code>[0,1&gt;</code></td>
</tr>
</tbody>
</table>

its default value of 10 as of Tracer

**Value**

the data frame with the burn-in removed

**Author(s)**

Richèl J.C. Bilderbeek
save_beast_estimates

Save the BEAST2 estimates as a BEAST2 .log file. There will be some differences: a BEAST2 .log file also saves the model as comments and formats the numbers in a way non-standard to R.

Usage

save_beast_estimates(estimates, filename)

Arguments

estimates a data frame of BEAST2 parameter estimates
filename name of the .log file to save to

Value

nothing

Author(s)

Richèl J.C. Bilderbeek

See Also

Use parse_beast_log to read a BEAST2 .log file

save_beast_trees

Save the BEAST2 trees as a BEAST2 .log file. There will be some differences: a BEAST2 .log file also saves the model as comments and formats the numbers in a way non-standard to R.

Usage

save_beast_trees(trees, filename)
save_beast_trees

Arguments

- `trees` BEAST2 posterior trees, of type `ape::multiPhylo`
- `filename` name of the `.trees` file to save to

Value

nothing

Author(s)

Richèl J.C. Bilderbeek

See Also

Use `parse_beast_log` to read a BEAST2 .log file
Index

calc_act, 3, 4
calc_act_cpp, 4
calc_act_r, 4
calc_ess, 5
calc_esses, 6
calc_geom_mean, 7
calc_hpd_interval, 7
calc_mode, 8
calc_std_error_of_mean_cpp, 10
calc_stderr_mean, 9
calc_summary_stats, 10
calc_summary_stats_trace, 11, 12
calc_summary_stats_traces, 11, 13
check_trace, 14
count_trees_in_file, 14
cs_std_dev, 15
default_params_doc, 15
extract_operators_lines, 16
get_tracerer_path, 17, 18
get_tracerer_paths, 17, 18
get_tracerer_tempfilename, 19
is_posterior, 19
is_trees_file, 14, 20, 26
is_trees_posterior, 21

parse_beast_log, 21, 28, 29
parse_beast_output_files, 22
parse_beast_posterior, 23
parse_beast_state_operators, 24
parse_beast_tracelog_file, 16, 21, 23, 25, 25
parse_beast_trees, 14, 16, 20, 22, 23, 26

read.nexus, 20
remove_burn_in, 8, 11, 12, 26
remove_burn_ins, 11, 13, 23–25, 27

save_beast_estimates, 25, 28
save_beast_trees, 26, 28
stop, 14
tempfile, 19
tracerer, 19