

Package ‘babette’

October 29, 2018

Title Control BEAST2 from R

Version 1.3

Author Richel J.C. Bilderbeek <richel@richelbilderbeek.nl>

Maintainer Richel J.C. Bilderbeek <richel@richelbilderbeek.nl>

Description 'BEAST2' (<<http://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. 'BEAST2' is commonly accompanied by 'BEAUti 2' (<<http://www.beast2.org>>), 'Tracer' (<<http://tree.bio.ed.ac.uk/software/tracer/>>) and DensiTree (<<http://www.beast2.org>>). 'babette' provides for an alternative workflow of using all these tools separately. This allows doing complex Bayesian phylogenetics easily and reproducibly from R.

License GPL-3 | file LICENSE

LazyData true

RoxygenNote 6.1.0

VignetteBuilder knitr

URL <https://github.com/richelbilderbeek/babette>

BugReports <https://github.com/richelbilderbeek/babette/issues>

Imports beautier, beastier, phangorn, stringr, tracerer, testit, xml2

Suggests ape, devtools, ggplot2, hunspell, knitr, lintr, mauricer, nLTT, rmarkdown, spelling, testthat

Remotes jimhester/covr, jimhester/lintr, richelbilderbeek/beautier, richelbilderbeek/beastier, richelbilderbeek/nLTT, richelbilderbeek/tracerer, MangoTheCat/goodpractice, KlausVigo/phangorn

Language en-US

Encoding UTF-8

NeedsCompilation no

R topics documented:

babette	2
bbt_check_beast2_packages	2
bbt_create_ns	3
bbt_default_params_doc	4
bbt_process_pkg_output	5
bbt_run	6
get_babette_path	9
get_babette_paths	10
plot_densitree	11
Index	12

babette	<i>babette: A package for Bayesian phylogenetics.</i>
---------	---

Description

'babette' provides for an alternative workflow of using the popular phylogenetics tool BEAST2, including it peripheral tools. From an alignment and prior selection, a posterior of jointly estimated phylogenies and parameter estimates is generated.

See Also

- These are packages associated with 'beautier':
- The package 'beautier' can create BEAST2 input files from R
 - The package 'beastier' can run BEAST2 from R
 - The package 'mauricer' can install BEAST2 packages from R
 - The package 'tracerer' can parse BEAST2 output files from R

bbt_check_beast2_packages	<i>Checks if bbt_run has the BEAST packages needed to process its arguments. Will throw if not.</i>
---------------------------	---

Description

Checks if bbt_run has the BEAST packages needed to process its arguments. Will throw if not.

Usage

```
bbt_check_beast2_packages(mcmc,  
  beast2_path = get_default_beast2_bin_path())
```

Arguments

mcmc	the MCMC options, see create_mcmc
beast2_path	name of either a BEAST2 binary file (usually simply beast) or a BEAST2 jar file (usually has a .jar extension). Use <code>get_default_beast2_bin_path</code> to get the default BEAST binary file's path Use <code>get_default_beast2_jar_path</code> to get the default BEAST jar file's path

bbt_create_ns	<i>Put the info of a Nested Sampling run in a structure</i>
---------------	---

Description

Put the info of a Nested Sampling run in a structure

Usage

```
bbt_create_ns(output)
```

Arguments

output	screen output
--------	---------------

Value

a list with the following elements:

- `marg_log_lik` the marginal log likelihood estimate
- `marg_log_lik_sd` the standard deviation around the estimate
- `ess` the effective sample size in estimating the marginal log likelihood

Author(s)

Richel J.C. Bilderbeek

bbt_default_params_doc

This function does nothing. It is intended to inherit is parameters' documentation.

Description

This function does nothing. It is intended to inherit is parameters' documentation.

Usage

```
bbt_default_params_doc(beast2_input_filename, beast2_output_log_filename,
    beast2_output_state_filename, beast2_output_trees_filenames, beast2_path,
    cleanup, clock_models, fasta_filenames, mcmc, mrca_priors,
    posterior_crown_age, rng_seed, site_models, tree_priors, verbose)
```

Arguments

- beast2_input_filename
path of the BEAST2 configuration file. By default, this file is put in a temporary folder with a random filename, as the user needs not read it: it is used as input of BEAST2. Specifying a beast2_input_filename allows to store that file in a more permanently stored location.
- beast2_output_log_filename
name of the log file created by BEAST2, containing the parameter estimates in time. By default, this file is put a temporary folder with a random filename, as the user needs not read it: its content is parsed and returned by this function. Specifying a beast2_output_log_filename allows to store that file in a more permanently stored location.
- beast2_output_state_filename
name of the final state file created by BEAST2, containing the operator acceptances. By default, this file is put a temporary folder with a random filename, as the user needs not read it: its content is parsed and returned by this function. Specifying a beast2_output_state_filename allows to store that file in a more permanently stored location.
- beast2_output_trees_filenames
name of the one or more trees files created by BEAST2, one per alignment. By default, these files are put a temporary folder with a random filename, as the user needs not read it: their content is parsed and returned by this function. Specifying beast2_output_trees_filenames allows to store these one or more files in a more permanently stored location.
- beast2_path
name of either a BEAST2 binary file (usually simply beast) or a BEAST2 jar file (usually has a .jar extension). Use get_default_beast2_bin_path to get the default BEAST binary file's path Use get_default_beast2_jar_path to get the default BEAST jar file's path
- cleanup
set to FALSE to keep all temporary files

clock_models	one or more clock models, see create_clock_models
fasta_filenames	one or more FASTA filename, each with one alignment
mcmc	the MCMC options, see create_mcmc
mrca_priors	a list of one or more Most Recent Common Ancestor priors, as returned by create_mrca_prior
posterior_crown_age	the posterior's crown age. Use NA to let BEAST2 estimate this parameter. Use a positive value to fix the crown age to that value
rng_seed	the random number generator seed. Must be either NA or a positive non-zero value. An RNG seed of NA results in BEAST2 picking a random seed.
site_models	one or more site models, see create_site_models
tree_priors	one or more tree priors, see create_tree_priors
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)

Richel J.C. Bilderbeek

bibt_process_pkg_output

Process the BEAST2 output dependent on BEAST2 package specifics

Description

Process the BEAST2 output dependent on BEAST2 package specifics

Usage

```
bbt_process_pkg_output(out, mcmc)
```

Arguments

out	complete babette output
mcmc	an MCMC

bbt_run	<i>Do a full run: create a BEAST2 configuration file (like BEAUti 2), run BEAST2, parse results (like Tracer)</i>
---------	---

Description

Do a full run: create a BEAST2 configuration file (like BEAUti 2), run BEAST2, parse results (like Tracer)

Usage

```
bbt_run(
  fasta_filenames,
  site_models = beautier::create_jc69_site_models(
    beautier::get_alignment_ids(fasta_filenames)
  ),
  clock_models = beautier::create_strict_clock_models(
    beautier::get_alignment_ids(fasta_filenames)
  ),
  tree_priors = beautier::create_yule_tree_priors(
    beautier::get_alignment_ids(fasta_filenames)
  ),
  mrca_priors = NA,
  mcmc = beautier::create_mcmc(),
  posterior_crown_age = NA,
  beast2_input_filename = tempfile(pattern = "beast2_", fileext = ".xml"),
  rng_seed = 1,
  beast2_output_log_filename = tempfile(
    pattern = "beast2_", fileext = "log"
  ),
  beast2_output_trees_filenames = tempfile(
    pattern = paste0(
      "beast2_",
      beautier::get_alignment_ids(fasta_filenames), "_"
    ),
    fileext = ".trees"
  ),
  beast2_output_state_filename = tempfile(
    pattern = "beast2_", fileext = ".xml.state"
  ),
  beast2_path = beautier::get_default_beast2_path(),
  verbose = FALSE,
  cleanup = TRUE
)
```

Arguments

fasta_filenames	one or more FASTA filename, each with one alignment
site_models	one or more site models, see create_site_models
clock_models	one or more clock models, see create_clock_models
tree_priors	one or more tree priors, see create_tree_priors
mrca_priors	a list of one or more Most Recent Common Ancestor priors, as returned by create_mrca_prior
mcmc	the MCMC options, see create_mcmc
posterior_crown_age	the posterior's crown age. Use NA to let BEAST2 estimate this parameter. Use a positive value to fix the crown age to that value
beast2_input_filename	path of the BEAST2 configuration file. By default, this file is put in a temporary folder with a random filename, as the user needs not read it: it is used as input of BEAST2. Specifying a <code>beast2_input_filename</code> allows to store that file in a more permanently stored location.
rng_seed	the random number generator seed. Must be either NA or a positive non-zero value. An RNG seed of NA results in BEAST2 picking a random seed.
beast2_output_log_filename	name of the log file created by BEAST2, containing the parameter estimates in time. By default, this file is put a temporary folder with a random filename, as the user needs not read it: its content is parsed and returned by this function. Specifying a <code>beast2_output_log_filename</code> allows to store that file in a more permanently stored location.
beast2_output_trees_filenames	name of the one or more trees files created by BEAST2, one per alignment. By default, these files are put a temporary folder with a random filename, as the user needs not read it: their content is parsed and returned by this function. Specifying <code>beast2_output_trees_filenames</code> allows to store these one or more files in a more permanently stored location.
beast2_output_state_filename	name of the final state file created by BEAST2, containing the operator acceptances. By default, this file is put a temporary folder with a random filename, as the user needs not read it: its content is parsed and returned by this function. Specifying a <code>beast2_output_state_filename</code> allows to store that file in a more permanently stored location.
beast2_path	name of either a BEAST2 binary file (usually simply <code>beast</code>) or a BEAST2 jar file (usually has a <code>.jar</code> extension). Use <code>get_default_beast2_bin_path</code> to get the default BEAST binary file's path Use <code>get_default_beast2_jar_path</code> to get the default BEAST jar file's path
verbose	set to TRUE for more output
cleanup	set to FALSE to keep all temporary files

Value

a list with the following elements:

- `estimates`: a data frame with BEAST2 parameter estimates
- `[alignment_id]_trees`: a multiPhylo containing the phylogenies in the BEAST2 posterior. `[alignment_id]` is the ID of the alignment. For example, when running `bbt_run` with `anthus_aco.fas`, this element will have name `anthus_aco_trees`
- `operators`: a data frame with the BEAST2 MCMC operator acceptances
- `output`: a numeric vector with the output sent to standard output and error streams
- `ns`: (optional) the results of the NS BEAST2 package, will exist only when `create_mcmc_nested_sampling` was used for mcmc

Author(s)

Richel J.C. Bilderbeek

See Also

Use [remove_burn_ins](#) to remove the burn-ins from the posterior's estimates (`posterior$estimates`)

Examples

```
# One alignment
out <- bbt_run(
  fasta_filenames = get_babette_path("anthus_aco.fas"),
  mcmc = create_mcmc(chain_length = 1000, store_every = 1000)
)

testit::assert("estimates" %in% names(out))
testit::assert("anthus_aco_trees" %in% names(out))
testit::assert("operators" %in% names(out))
testit::assert("output" %in% names(out))
testit::assert(class(out$anthus_aco_trees[[1]]) == "phylo")
testit::assert(length(out$anthus_aco_trees) == 2)

testit::assert("Sample" %in% names(out$estimates))
testit::assert("posterior" %in% names(out$estimates))
testit::assert("likelihood" %in% names(out$estimates))
testit::assert("prior" %in% names(out$estimates))
testit::assert("treeLikelihood" %in% names(out$estimates))
testit::assert("TreeHeight" %in% names(out$estimates))
testit::assert("YuleModel" %in% names(out$estimates))
testit::assert("birthRate" %in% names(out$estimates))

testit::assert("operator" %in% names(out$operators))
testit::assert("p" %in% names(out$operators))
testit::assert("accept" %in% names(out$operators))
testit::assert("reject" %in% names(out$operators))
testit::assert("acceptFC" %in% names(out$operators))
```



```

testit::assert("rejectFC" %in% names(out$operators))
testit::assert("rejectIv" %in% names(out$operators))
testit::assert("rejectOp" %in% names(out$operators))

# Two alignments
out <- bbt_run(
  fasta_filenames = get_babette_paths(
    c("anthus_aco.fas", "anthus_nd2.fas")
  ),
  mcmc = create_mcmc(chain_length = 1000, store_every = 1000)
)

testit::assert("estimates" %in% names(out))
testit::assert("anthus_aco_trees" %in% names(out))
testit::assert("anthus_nd2_trees" %in% names(out))
testit::assert("operators" %in% names(out))
testit::assert(class(out$anthus_aco_trees[[1]]) == "phylo")
testit::assert(class(out$anthus_nd2_trees[[1]]) == "phylo")
testit::assert(length(out$anthus_aco_trees) == 2)
testit::assert(length(out$anthus_nd2_trees) == 2)

testit::assert("Sample" %in% names(out$estimates))
testit::assert("posterior" %in% names(out$estimates))
testit::assert("likelihood" %in% names(out$estimates))
testit::assert("prior" %in% names(out$estimates))
testit::assert("treeLikelihood.aco" %in% names(out$estimates))
testit::assert("treeLikelihood.nd2" %in% names(out$estimates))
testit::assert("TreeHeight.aco" %in% names(out$estimates))
testit::assert("TreeHeight.nd2" %in% names(out$estimates))
testit::assert("YuleModel.aco" %in% names(out$estimates))
testit::assert("YuleModel.nd2" %in% names(out$estimates))
testit::assert("birthRate.aco" %in% names(out$estimates))
testit::assert("birthRate.nd2" %in% names(out$estimates))

```

get_babette_path

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

Description

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

Usage

```
get_babette_path(filename)
```

Arguments

filename the file's name, without the path

Value

the full path of the filename, if and only if the file is present. Will stop otherwise.

Author(s)

Richel J.C. Bilderbeek

See Also

for more files, use [get_babette_paths](#)

Examples

```
testit::assert(is.character(get_babette_path("anthus_aco.fas")))
testit::assert(is.character(get_babette_path("anthus_nd2.fas")))
```

get_babette_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

```
get_babette_paths(filenamees)
```

Arguments

filenamees the files' names, without the path

Value

the filenamees' full paths, if and only if all files are present. Will stop otherwise.

Author(s)

Richel J.C. Bilderbeek

See Also

for one file, use [get_babette_path](#)

Examples

```
testit::assert(
  length(
    get_babette_paths(
      c("anthus_aco.fas", "anthus_nd2.fas")
    )
  ) == 2
)
```

plot_densitree	<i>Calls phangorn's densiTree function</i>
----------------	--

Description

Calls phangorn's [densiTree](#) function

Usage

```
plot_densitree(phylos, ...)
```

Arguments

phylos	one or more phylogenies, must be of class multiPhylo
...	options to be passed to phangorn's densiTree function

Value

nothing. Will produce a plot.

Author(s)

Richel J.C. Bilderbeek

Examples

```
out <- bbt_run(
  get_babette_path("anthus_aco.fas"),
  mcmc = create_mcmc(chain_length = 10000, store_every = 1000)
)
plot_densitree(out$anthus_aco_trees)
```

Index

`babette`, [2](#)
`babette-package (babette)`, [2](#)
`bbt_check_beast2_packages`, [2](#)
`bbt_create_ns`, [3](#)
`bbt_default_params_doc`, [4](#)
`bbt_process_pkg_output`, [5](#)
`bbt_run`, [6](#)

`create_clock_models`, [5](#), [7](#)
`create_mcmc`, [3](#), [5](#), [7](#)
`create_mrca_prior`, [5](#), [7](#)
`create_site_models`, [5](#), [7](#)
`create_tree_priors`, [5](#), [7](#)

`densiTree`, [11](#)

`get_babette_path`, [9](#), [10](#)
`get_babette_paths`, [10](#), [10](#)

`plot_densitree`, [11](#)

`remove_burn_ins`, [8](#)