Package ‘RoBTT’

September 20, 2023

Title  Robust Bayesian T-Test
Version  1.2.1
Maintainer  František Bartoš <f.bartos96@gmail.com>

Description  An implementation of Bayesian model-averaged t-test that allows
users to draw inference about the presence vs absence of the effect,
heterogeneity of variances, and outliers. The ‘RoBTT’ packages estimates model
ensembles of models created as a combination of the competing hypotheses and uses
Bayesian model-averaging to combine the models using posterior model probabilities.
Users can obtain the model-averaged posterior distributions and inclusion Bayes
factors which account for the uncertainty in the data generating process
(Maier et al., 2022, <doi:10.31234/osf.io/d5zwc>).
Users can define a wide range of informative priors for all parameters
of interest. The package provides convenient functions for summary, visualizations,
and fit diagnostics.

URL  https://fbartos.github.io/RoBTT/
BugReports  https://github.com/FBartos/RoBTT/issues
License  GPL-3
Encoding  UTF-8
LazyData  true
SystemRequirements  GNU make
RoxygenNote  7.2.3
NeedsCompilation  yes
ByteCompile  true

LinkingTo  StanHeaders (>= 2.26.0), rstan (>= 2.26.0), BH (>= 1.69.0),
             Rcpp (>= 0.12.15), RcppEigen (>= 0.3.3.4.0), RcppParallel (>=
             5.0.1)

Depends  R (>= 4.0.0), Rcpp (>= 0.12.19)
Imports  rstan(>= 2.26.0), rstantools(>= 1.5.0), RcppParallel (>=
             5.0.1), BayesTools (>= 0.2.15), bridgesampling, methods,
ggplot2, Rdpack
RoBTT-package

Suggests parallel, testthat, vdiffr, knitr, rmarkdown, covr

RdMacros Rdpack

VignetteBuilder knitr

Author František Bartoš [aut, cre] (<https://orcid.org/0000-0002-0018-5573>), Maximilian Maier [aut] (<https://orcid.org/0000-0002-9873-6096>)

Repository CRAN

Date/Publication 2023-09-20 08:30:02 UTC

**R topics documented:**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>RoBTT-package</td>
<td>2</td>
</tr>
<tr>
<td>check_RoBTT</td>
<td>3</td>
</tr>
<tr>
<td>check_setup</td>
<td>4</td>
</tr>
<tr>
<td>diagnostics</td>
<td>5</td>
</tr>
<tr>
<td>fertilization</td>
<td>7</td>
</tr>
<tr>
<td>interpret</td>
<td>8</td>
</tr>
<tr>
<td>is.RoBTT</td>
<td>8</td>
</tr>
<tr>
<td>plot.RoBTT</td>
<td>9</td>
</tr>
<tr>
<td>print.RoBTT</td>
<td>10</td>
</tr>
<tr>
<td>print.summary.RoBTT</td>
<td>11</td>
</tr>
<tr>
<td>prior</td>
<td>11</td>
</tr>
<tr>
<td>prior_none</td>
<td>13</td>
</tr>
<tr>
<td>rho2logsdr</td>
<td>14</td>
</tr>
<tr>
<td>RoBTT</td>
<td>14</td>
</tr>
<tr>
<td>RoBTT_control</td>
<td>17</td>
</tr>
<tr>
<td>RoBTT_options</td>
<td>18</td>
</tr>
<tr>
<td>summary.RoBTT</td>
<td>18</td>
</tr>
</tbody>
</table>

Index 21

**Description**

RoBTT: Bayesian model-averaged t-test extends the Bayesian t-test by incorporating inference about heterogeneity of variances and outliers.

**User guide**

See Maier et al. (2022) for details regarding the RoBTT methodology.

More details regarding customization of the Bayesian model-averaged t-test are provided in Introduction to RoBTT vignette. Please, use the "Issues" section in the GitHub repository to ask any further questions.
check_RoBTT

**Author(s)**

František Bartoš <f.bartos96@gmail.com>

**References**


**See Also**

Useful links:

- https://fbartos.github.io/RoBTT/
- Report bugs at https://github.com/FBartos/RoBTT/issues

---

**check_RoBTT**  
*Check fitted 'RoBTT' object for errors and warnings*

**Description**

Checks fitted 'RoBTT' object for warnings and errors and prints them to the console.

**Usage**

`check_RoBTT(fit)`

**Arguments**

- `fit`  
a fitted 'RoBTT' object.

**Value**

`check_RoBTT` returns a vector of error and warning messages.
Description

check_setup prints summary of "RoBTT" ensemble implied by the specified prior distributions. It is useful for checking the ensemble configuration prior to fitting all of the models.

Usage

```r
check_setup(
  prior_delta = prior(distribution = "cauchy", parameters = list(location = 0, scale = sqrt(2)/2)),
  prior_rho = prior(distribution = "beta", parameters = list(alpha = 1, beta = 1)),
  prior_nu = prior(distribution = "exp", parameters = list(rate = 1)),
  prior_delta_null = prior(distribution = "spike", parameters = list(location = 0)),
  prior_rho_null = prior(distribution = "spike", parameters = list(location = 0.5)),
  prior_nu_null = prior_none(),
  models = FALSE,
  silent = FALSE
)
```

Arguments

- `prior_delta`: prior distributions for the effect size delta parameter that will be treated as belonging to the alternative hypothesis. Defaults to `prior(distribution = "Cauchy", parameters = list(location = 0, scale = sqrt(2)/2))`.
- `prior_rho`: prior distributions for the precision allocation rho parameter that will be treated as belonging to the alternative hypothesis. Defaults to `prior(distribution = "beta", parameters = list(alpha = 1, beta = 1))`.
- `prior_nu`: prior distribution for the degrees of freedom + 2 nu parameter that will be treated as belonging to the alternative hypothesis. Defaults to `prior(distribution = "exp", parameters = list(rate = 1))`.
- `prior_delta_null`: prior distribution for the delta parameter that will be treated as belonging to the null hypothesis. Defaults to point distribution with location at 0 (`prior(distribution = "point", parameters = list(location = 0))`).
- `prior_rho_null`: prior distribution for the rho parameter that will be treated as belonging to the null hypothesis. Defaults to point distribution with location at 0.5 (`prior(distribution = "point", parameters = list(location = 0.5))`).
- `prior_nu_null`: prior distribution for the nu parameter that will be treated as belonging to the null hypothesis. Defaults to `prior_none` (i.e., normal likelihood).
- `models`: should the models' details be printed.
- `silent`: do not print the results.
Value

check_setup invisibly returns list of summary tables.

See Also

RoBTT()

diagnostics  

**Checks a fitted RoBTT object**

Description

diagnostics creates visual checks of individual models convergence. Numerical overview of individual models can be obtained by `summary(object, type = "models", diagnostics = TRUE)`, or even more detailed information by `summary(object, type = "individual")`.

Usage

diagnostics(
  fit,  
  parameter,  
  type,  
  show_models = NULL,  
  lags = 30,  
  title = is.null(show_models) | length(show_models) > 1,  
  ...  
)

diagnostics_autocorrelation(
  fit,  
  parameter = NULL,  
  show_models = NULL,  
  lags = 30,  
  title = is.null(show_models) | length(show_models) > 1,  
  ...  
)

diagnostics_trace(
  fit,  
  parameter = NULL,  
  show_models = NULL,  
  title = is.null(show_models) | length(show_models) > 1,  
  ...  
)

diagnostics_density(
  fit,
  ...
diagnostics

parameter = NULL,
show_models = NULL,
title = is.null(show_models) | length(show_models) > 1,

Arguments

fit a fitted RoBTT object
parameter a parameter to be plotted. Either "delta", "rho", "nu", "mu", or "sigma".
type type of MCMC diagnostic to be plotted. Options are "chains" for the chains' trace plots, "autocorrelation" for autocorrelation of the chains, and "densities" for the overlaying densities of the individual chains. Can be abbreviated to first letters.
show_models MCMC diagnostics of which models should be plotted. Defaults to NULL which plots MCMC diagnostics for a specified parameter for every model that is part of the ensemble.
lags number of lags to be shown for type = "autocorrelation". Defaults to 30.
title whether the model number should be displayed in title. Defaults to TRUE when more than one model is selected.

Details

The visualization functions are based on stan_plot function and its color schemes.

Value

diagnostics returns either NULL if plot_type = "base" or an object/list of objects (depending on the number of parameters to be plotted) of class 'ggplot2' if plot_type = "ggplot2".

See Also

RoBTT(), summary.RoBTT()

Examples

```r
## Not run:
# using the example data from Darwin
data("fertilization", package = "RoBTT")
fit <- RoBTT(
  x1 = fertilization$Self,
  x2 = fertilization$Crossed,
  prior_delta = prior("cauchy", list(0, 1/sqrt(2))),
  prior_rho = prior("beta", list(3, 3)),
  seed = 1,
  chains = 1,
  warmup = 1000,
  iter = 2000,
  ...)
```
control = set_control(adapt_delta = 0.95)
)

# diagnostics function allows to visualize diagnostics of a fitted RoBTT object, for example,
# the trace plot for the mean parameter in each model model
# in order to show the trace plot only for the 11th model, add show_models parameter
diagnostics(fit, parameter = "delta", type = "chain")

diagnostics(fit, parameter = "delta", type = "chain", show_models = 11)

# furthermore, the autocorrelations
diagnostics(fit, parameter = "delta", type = "autocorrelation")

# and overlying densities for each plot can also be visualize
diagnostics(fit, parameter = "delta", type = "densities")

## End(Not run)

---

**fertilization**

*Height of 15 plant pairs collected by Darwin*

---

**Description**

The data set contains heights of self fertilized and cross fertilized plants as coded by Darwin.

**Usage**

fertilization

**Format**

A data.frame with 2 columns and 15 observations.

**Value**

a data.frame.
interpret

Interprets results of a 'RoBTT' model.

**Description**

interpret creates a brief textual summary of a fitted 'RoBTT' object.

**Usage**

interpret(object)

**Arguments**

object a fitted 'RoBTT' object

**Value**

interpret returns a character.

is.RoBTT

Reports whether x is a 'RoBTT' object

**Description**

Reports whether x is a 'RoBTT' object

**Usage**

is.RoBTT(x)

**Arguments**

x an object to test

**Value**

is.RoBTT returns a boolean.
### Description

`plot.RoBTT` allows to visualize different "RoBTT" object parameters in various ways. See `type` for the different model types.

### Usage

```r
## S3 method for class 'RoBTT'
plot(
  x,
  parameter = "mu",
  transform_rho = FALSE,
  conditional = FALSE,
  plot_type = "base",
  prior = FALSE,
  dots_prior = NULL,
  ...
)
```

### Arguments

- **x**: a fitted 'RoBTT' object
- **parameter**: a parameter to be plotted. Defaults to "delta" (for the effect size). The additional options are "rho" (for the heterogeneity), "nu" (for the degrees of freedom).
- **transform_rho**: whether rho parameter should be translated into log standard deviation ratio
- **conditional**: whether conditional estimates should be plotted. Defaults to `FALSE` which plots the model-averaged estimates.
- **plot_type**: whether to use a base plot "base" or ggplot2 "gplot" for plotting. Defaults to "base".
- **prior**: whether prior distribution should be added to figure. Defaults to `FALSE`.
- **dots_prior**: list of additional graphical arguments to be passed to the plotting function of the prior distribution. Supported arguments are `lwd`, `lty`, `col`, and `col.fill`, to adjust the line thickness, line type, line color, and fill color of the prior distribution respectively.
- **...**: list of additional graphical arguments to be passed to the plotting function. Supported arguments are `lwd`, `lty`, `col`, `col.fill`, `xlab`, `ylab`, `main`, `xlim`, `ylim` to adjust the line thickness, line type, line color, fill color, x-label, y-label, title, x-axis range, and y-axis range respectively.
plot.RoBTT returns either `NULL` if `plot_type = "base"` or an object object of class `"ggplot2"` if `plot_type = "ggplot2"`.

See Also

RoBTT()

Examples

```r
## Not run:
data("fertilization", package = "RoBTT")
fit <- RoBTT(
  x1 = fertilization$Self,
  x2 = fertilization$Crossed,
  prior_delta = prior("cauchy", list(0, 1/sqrt(2))),
  prior_rho = prior("beta", list(3, 3)),
  seed = 1,
  chains = 1,
  warmup = 1000,
  iter = 2000,
  control = set_control(adapt_delta = 0.95)
)

# plot the model-averaged effect size estimate
plot(fit, parameter = "delta")

# plot prior and posterior of the conditional effect size estimate
plot(fit, parameter = "delta", conditional = TRUE, prior = TRUE)
## End(Not run)
```

Description

Prints a fitted 'RoBTT' object

Usage

```r
## S3 method for class 'RoBTT'
print(x, ...)
```

Arguments

- `x`: a fitted 'RoBTT' object.
- `...`: additional arguments.
**Value**

print.RoBTT invisibly returns the print statement.

**See Also**

RoBTT()

---

**Description**

Prints summary object for 'RoBTT' method

**Usage**

```r
## S3 method for class 'summary.RoBTT'
print(x, ...)  
```

**Arguments**

- `x`: a summary of a 'RoBTT' object
- `...`: additional arguments

**Value**

print.summary.RoBTT invisibly returns the print statement.

**See Also**

RoBTT()

---

**Description**

prior creates a prior distribution. The prior can be visualized by the plot function.

**Usage**

```r
prior(
  distribution,
  parameters,
  truncation = list(lower = -Inf, upper = Inf),
  prior_weights = 1
)
```
Arguments

distribution  name of the prior distribution. The possible options are
"point"  for a point density characterized by a location parameter.
"normal"  for a normal distribution characterized by a mean and sd parameters.
"lognormal"  for a lognormal distribution characterized by a meanlog and sdlog parameters.
"cauchy"  for a Cauchy distribution characterized by a location and scale parameters. Internally converted into a generalized t-distribution with df = 1.
"t"  for a generalized t-distribution characterized by a location, scale, and df parameters.
"gamma"  for a gamma distribution characterized by either shape and rate, or shape and scale parameters. The later is internally converted to the shape and rate parametrization.
"invgamma"  for an inverse-gamma distribution characterized by a shape and scale parameters. The JAGS part uses a 1/gamma distribution with a shape and rate parameter.
"beta"  for a beta distribution characterized by an alpha and beta parameters.
"exp"  for an exponential distribution characterized by either rate or scale parameter. The later is internally converted to rate.
"uniform"  for a uniform distribution defined on a range from a to b

parameters  list of appropriate parameters for a given distribution.

truncation  list with two elements, lower and upper, that define the lower and upper truncation of the distribution. Defaults to list(lower = -Inf, upper = Inf). The truncation is automatically set to the bounds of the support.

prior_weights  prior odds associated with a given distribution. The value is passed into the model fitting function, which creates models corresponding to all combinations of prior distributions for each of the model parameters and sets the model priors odds to the product of its prior distributions.

Value

prior and prior_none return an object of class 'prior'. A named list containing the distribution name, parameters, and prior weights.

See Also

plot.prior(), Normal, Lognormal, Cauchy, Beta, Exponential, LocationScaleT, InvGamma.

Examples

# create a standard normal prior distribution
p1 <- prior(distribution = "normal", parameters = list(mean = 1, sd = 1))

# create a half-normal standard normal prior distribution
p2 <- prior(distribution = "normal", parameters = list(mean = 1, sd = 1)},
prior_none

truncation = list(lower = 0, upper = Inf))

# the prior distribution can be visualized using the plot function
# (see ?plot.prior for all options)
plot(p1)

prior_none  Creates a prior distribution

Description

prior creates a prior distribution. The prior can be visualized by the plot function.

Usage

prior_none(prior_weights = 1)

Arguments

prior_weights  prior odds associated with a given distribution. The value is passed into the model fitting function, which creates models corresponding to all combinations of prior distributions for each of the model parameters and sets the model priors odds to the product of its prior distributions.

Value

prior and prior_none return an object of class 'prior'. A named list containing the distribution name, parameters, and prior weights.

See Also

plot.prior(), Normal, Lognormal, Cauchy, Beta, Exponential, LocationScaleT, InvGamma.

Examples

# create a standard normal prior distribution
p1 <- prior(distribution = "normal", parameters = list(mean = 1, sd = 1))

# create a half-normal standard normal prior distribution
p2 <- prior(distribution = "normal", parameters = list(mean = 1, sd = 1),
truncation = list(lower = 0, upper = Inf))

# the prior distribution can be visualized using the plot function
# (see ?plot.prior for all options)
plot(p1)
rho2logsdr  
*rho to log standard deviation ratio transformations*

**Description**

A list containing the transformation function, inverse transformation function, and the jacobian function.

**Usage**

rho2logsdr

**Format**

An object of class list of length 3.

**Value**

a list with the corresponding functions

---

RoBTT  
*Estimate a Robust Bayesian T-Test*

**Description**

RoBTT is used to estimate a Robust Bayesian T-Test. The input either requires the vector of observations for each group, \( x_1, x_2 \), or the summary statistics (in case only the "normal" likelihood is used).

**Usage**

RoBTT(  
  x1 = NULL,  
  x2 = NULL,  
  mean1 = NULL,  
  mean2 = NULL,  
  sd1 = NULL,  
  sd2 = NULL,  
  N1 = NULL,  
  N2 = NULL,  
  prior_delta = prior(distribution = "cauchy", parameters = list(location = 0, scale = sqrt(2)/2)),  
  prior_rho = prior(distribution = "beta", parameters = list(alpha = 1, beta = 1)),  
  prior_nu = prior(distribution = "exp", parameters = list(rate = 1)),  
  prior_delta_null = prior(distribution = "spike", parameters = list(location = 0)),  
  prior_rho_null = prior(distribution = "spike", parameters = list(location = 0.5)))
prior_nu_null = prior_none(),
chains = 4,
iter = 10000,
warmup = 5000,
thin = 1,
parallel = FALSE,
control = set_control(),
convergence_checks = set_convergence_checks(),
save = "all",
seed = NULL,
silent = TRUE,
...
)

Arguments

x1 vector of observations of the first group
x2 vector of observations of the second group
mean1 mean of the first group
mean2 mean of the first group
sd1 standard deviation of the first group
sd2 standard deviation of the first group
N1 sample size of the first group
N2 sample size of the first group
prior_delta prior distributions for the effect size delta parameter that will be treated as belonging to the alternative hypothesis. Defaults to prior(distribution = "Cauchy", parameters = list(location = 0, scale = sqrt(2)/2)).
prior_rho prior distributions for the precision allocation rho parameter that will be treated as belonging to the alternative hypothesis. Defaults to prior(distribution = "beta", parameters = list(alpha = 1, beta = 1)).
prior_nu prior distribution for the degrees of freedom + 2 nu parameter that will be treated as belonging to the alternative hypothesis. Defaults to prior(distribution = "exp", parameters = list(rate = 1)).
prior_delta_null prior distribution for the delta parameter that will be treated as belonging to the null hypothesis. Defaults to point distribution with location at 0 (prior(distribution = "point", parameters = list(location = 0))).
prior_rho_null prior distribution for the rho parameter that will be treated as belonging to the null hypothesis. Defaults to point distribution with location at 0.5 (prior(distribution = "point", parameters = list(location = 0.5))).
prior_nu_null prior distribution for the nu parameter that will be treated as belonging to the null hypothesis. Defaults to prior_none (i.e., normal likelihood).
chains a number of chains of the MCMC algorithm.
iter a number of sampling iterations of the MCMC algorithm. Defaults to 10000, with a minimum of 4000.
RoBTT

warmup a number of warmup iterations of the MCMC algorithm. Defaults to 5000.
thin a thinning of the chains of the MCMC algorithm. Defaults to 1.
parallel whether the individual models should be fitted in parallel. Defaults to FALSE.
The implementation is not completely stable and might cause a connection error.
control allows to pass control settings with the set_control() function. See ?set_control for options and default settings.
convergence_checks automatic convergence checks to assess the fitted models, passed with set_convergence_checks() function. See ?set_convergence_checks for options and default settings.
save whether all models posterior distributions should be kept after obtaining a model-averaged result. Defaults to "all" which does not remove anything. Set to "min" to significantly reduce the size of final object, however, some model diagnostics and further manipulation with the object will not be possible.
seed a seed to be set before model fitting, marginal likelihood computation, and posterior mixing for reproducibility of results. Defaults to NULL - no seed is set.
silent whether all print messages regarding the fitting process should be suppressed. Defaults to TRUE. Note that parallel = TRUE also suppresses all messages.
...
additional arguments.

Details

See Maier et al. (2022) for more details regarding the methodology.

Generic summary.RoBTT(), print.RoBTT(), and plot.RoBTT() functions are provided to facilitate manipulation with the ensemble.

Value

RoBTT returns an object of class "RoBTT".

References


See Also

summary.RoBTT(), prior()

Examples

## Not run:
# using the example data from Darwin
data("fertilization", package = "RoBTT")
fit <- RoBTT(
  x1 = fertilization$Self,
  x2 = fertilization$Crossed,
  prior_delta = prior("cauchy", list(0, 1/sqrt(2))),
  prior_rho = prior("beta", list(3, 3)),
...
seed = 1,
chains = 1,
warmup = 1000,
iter = 2000,
control = set_control(adapt_delta = 0.95)
)

# summary can provide many details about the model
summary(fit)

## End(Not run)

---

**Description**

Set values for the convergence checks of the fitting process.

**Usage**

```r
set_convergence_checks(max_Rhat = 1.05, min_ESS = 500)
set_control(adapt_delta = 0.8, max_treedepth = 15, bridge_max_iter = 1000)
```

**Arguments**

- `max_Rhat`: maximum value of the R-hat diagnostic. Defaults to 1.05.
- `min_ESS`: minimum estimated sample size. Defaults to 500.
- `adapt_delta`: tuning parameter of HMC. Defaults to 0.80.
- `max_treedepth`: tuning parameter of HMC. Defaults to 15.
- `bridge_max_iter`: maximum number of iterations for the `bridge_sampler` function. Defaults to 10000

**Value**

`set_control` returns a list of control settings and `set_convergence_checks` returns a list of convergence checks settings.
RoBTT_options

Options for the 'RoBTT' package

Description
A placeholder object and functions for the 'RoBTT' package. (adapted from the runjags R package).

Usage
RoBTT.options(...)

RoBTT.get_option(name)

Arguments
... named option(s) to change - for a list of available options, see details below.
name the name of the option to get the current value of - for a list of available options, see details below.

Value
The current value of all available 'RoBTT' options (after applying any changes specified) is returned invisibly as a named list.

summary.RoBTT

Summarize fitted 'RoBTT' object

Description
summary.RoBTT creates summary tables for a RoBTT object.

Usage
## S3 method for class 'RoBTT'
summary(
  object,
  type = "ensemble",
  conditional = FALSE,
  group_estimates = FALSE,
  probs = c(0.025, 0.975),
  logBF = FALSE,
  BF01 = FALSE,
  short_name = FALSE,
  remove_spike_0 = FALSE,
  ...
)
)
Arguments

- **object**: a fitted 'RoBTT' object
- **type**: whether to show the overall 'RoBTT' results ("ensemble"), an overview of the individual models ("models"), an overview of the individual models MCMC diagnostics ("diagnostics"), or a detailed summary of the individual models ("individual"). Can be abbreviated to first letters.
- **conditional**: show the conditional estimates (assuming that the alternative is true). Defaults to FALSE. Only available for type == "conditional".
- **group_estimates**: show the model-averaged mean and standard deviation estimates for each group.
- **probs**: quantiles of the posterior samples to be displayed. Defaults to c(.025, .975)
- **logBF**: show log of Bayes factors. Defaults to FALSE.
- **BF01**: show Bayes factors in support of the null hypotheses. Defaults to FALSE.
- **short_name**: whether priors names should be shortened to the first (couple) of letters. Defaults to FALSE.
- **remove_spike_0**: whether spike prior distributions with location at zero should be omitted from the summary. Defaults to FALSE.
- **...**: additional arguments

Value

`summary.RoBTT` returns a list of tables of class 'BayesTools_table'.

See Also

`RoBTT()`

Examples

```r
# Not run:
# using the example data from Darwin
data("fertilization", package = "RoBTT")
fit <- RoBTT(
  x1 = fertilization$Self,
  x2 = fertilization$Crossed,
  prior_delta = prior("cauchy", list(0, 1/sqrt(2))),
  prior_rho = prior("beta", list(3, 3)),
  seed = 1,
  chains = 1,
  warmup = 1000,
  iter = 2000,
  control = set_control(adapt_delta = 0.95)
)

# summary can provide many details about the model
summary(fit)
```
# estimates from the conditional models can be obtained with
summary(fit, conditional = TRUE)

# overview of the models and their prior and posterior probability, marginal likelihood,
# and inclusion Bayes factor can be obtained with
summary(fit, type = "models")

# diagnostics overview, containing the maximum R-hat, minimum ESS, maximum MCMC error, and
# maximum MCMC error / sd across parameters for each individual model can be obtained with
summary(fit, type = "diagnostics")

# summary of individual models and their parameters can be further obtained by
summary(fit, type = "individual")

## End(Not run)
Index

* datasets
  fertilization, 7
  rho2logsdr, 14

* package
  RoBTT-package, 2
  _PACKAGE (RoBTT-package), 2
  Beta, 12, 13
  bridge_sampler, 17
  Cauchy, 12, 13
  check_RoBTT, 3
  check_setup, 4
  class, 16
  diagnostics, 5
  diagnostics_autocorrelation (diagnostics), 5
  diagnostics_density (diagnostics), 5
  diagnostics_trace (diagnostics), 5
  Exponential, 12, 13
  fertilization, 7
  interpret, 8
  InvGamma, 12, 13
  is.RoBTT, 8
  LocationScaleT, 12, 13
  Lognormal, 12, 13
  Normal, 12, 13
  par, 6
  plot.prior(), 12, 13
  plot.RoBTT, 9
  plot.RoBTT(), 16
  print.RoBTT, 10
  print.RoBTT(), 16
  print.summary.RoBTT, 11
  prior, 11
  prior(), 16
  prior_none, 13
  rho2logsdr, 14
  RoBTT, 14
  RoBTT(), 5, 6, 10, 11, 19
  RoBTT-package, 2
  RoBTT.get_option (RoBTT_options), 18
  RoBTT.options (RoBTT_options), 18
  RoBTT.package (RoBTT-package), 2
  RoBTT_control, 17
  RoBTT_options, 18
  RoBTT.package (RoBTT-package), 2
  set_control (RoBTT_control), 17
  set_control(), 16
  set_control,(RoBTT_control), 17
  set_convergence_checks (RoBTT_control), 17
  set_convergence_checks(), 16
  stan_plot, 6
  summary.RoBTT, 18
  summary.RoBTT(), 6, 16