Package ‘BayesVolcano’

May 6, 2026

Title Creating Volcano Plots from Bayesian Model Posteriors

Version 1.0.1

URL `https://github.com/KatjaDanielzik/BayesVolcano`

BugReports `https://github.com/KatjaDanielzik/BayesVolcano/issues`

Description Bayesian models are used to estimate effect sizes (e.g., gene expression changes, protein abundance differences, drug response effects) while accounting for uncertainty, small sample sizes, and complex experimental designs.
However, Bayesian posteriors of models with many parameters are often difficult to interpret at a glance.
One way to quickly identify important biological changes based on frequentist analysis are volcano plots (using fold-changes and p-values).
Bayesian volcano plots bring together the explicit treatment of uncertainty in Bayesian models and the familiar visualization of volcano plots.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.5)

Imports ggplot2, HDInterval, purrr, dplyr, magrittr, tidyr

Suggests brms, rstan, knitr, rmarkdown, testthat (>= 3.0.0)

LazyData true

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Katja Danielzik [aut, cre, cph] (ORCID: <`https://orcid.org/0009-0007-5021-6212`>),
Simo Kitanovski [aut, ctb] (ORCID: <`https://orcid.org/0000-0003-2909-5376`>),
Daniel Hoffmann [aut] (ORCID: <`https://orcid.org/0000-0003-2973-7869`>)

Maintainer Katja Danielzik <`katja.danielzik@uni-due.de`>

Repository CRAN

Date/Publication 2026-03-31 09:50:15 UTC

Contents

`annotation_df` *A posterior with parameter "delta_mu"*

Description

A data with two columns "parameter" and "label" corresponding to data("posterior")

Usage

```
data("posterior")
```

Format

"Parameter" refers to column names of posterior "Lable" are the biological entities corresponding to the parameters

Source

Script used to create simulated data #BiocManager::install("MetaboDynamics") library(MetaboDynamics) data("longitudinalMetabolomics_df") library(BayesVolcano) data("posterior")

annotation_df <- as.data.frame(cbind(parameter=colnames(posterior), label=levels(as.factor(longitudinalMetabolomics_df$r
group=rep(c("A","B")),value=rnorm(98,10,3))) annotation_df$value <- as.numeric(annotation_df$value)

`extract_fit` *extract_fit*

Description

Wrapper function to extract parameter draws from two common Stan interfaces. This function requires the respective stan interface (rstan, brms) package to be installed.

Usage

```
extract_fit(fit, parameter_name)
```

Arguments

`fit`	A fitted Stan model object (stanfit, brmsfit).
`parameter_name`	A character string of parameter name

Value

A data frame with one row per MCMC draw and one column per parameter. If multiple parameters, columns are named after the parameter.

Examples

```
# Not run:
# fit <- brms::brm(count ~ zAge + zBase * Trt + (1|patient),
#             data = brms::epilepsy[1:30,], family = poisson())

# posterior <- extract_stan_fit(fit, "b_Intercept")
# End(Not run)
```

`plot_volcano` *Plot Bayesian Volcano plot*

Description

Plot Bayesian Volcano plot

Usage

```
plot_volcano(result, CrI = FALSE, CrI_width = FALSE, color = NULL)
```

Arguments

`result`	from `prepare_volcano_input()` (a data frame).
`CrI`	Logical. Whether to display the CrI Interval of the parameter
`CrI_width`	Logical. Whether to display the CrI width as point size.
`color`	Column in ‘result$result. Can be numerical or character.

Value

a ggplot2 object

Examples

```
data("posterior")
head(posterior)
data("annotation_df")
head(annotation_df)

result <- prepare_volcano_input(
  posterior = posterior,
  annotation = annotation_df,
)
plot_volcano(result,
  color = "group",
  CrI = TRUE,
  CrI_width = TRUE
)
```

`posterior` *A posterior with parameter "delta_mu"*

Description

A data frame with 98 columns (each indicating posterior samples of the paramter "delta_mu" of one biological entity like e.g. a metabolite) and 1000 rows (the number of total samples (chains*iterations)) of the Bayesian model.

Usage

```
data("posterior")
```

Format

A data frame with 98 columns (each indicating posterior samples of the paramter "delta_mu" of one biological entity like e.g. a metabolite) and 1000 rows (the number of total samples (chains*iterations)) of the Bayesian model.

Source

Script used to create simulated data #BiocManager::install("MetaboDynamics") library(MetaboDynamics) data("longitudinalMetabolomics_df") fit_dynamics_model(data=longitudinalMetabolomics_df) posterior <- as.data.frame(rstan::extract(fit,pars="euclidean_distance"))

library(stringr) cols <- colnames(posterior)[str_detect(pattern = ".1.2",string=colnames(posterior))] posterior <- posterior[,cols] posterior <- posterior[,-c(1:10,109:118)]

`prepare_volcano_input` *Prepare volcano input*

Description

This function has as input posterior draws, calculates pi-values and credible intervals (CrI), and annotates them with biological information (e.g., cell line, time point) based on parameter names and a user-provided annotation data frame. Returns a data frame that is ready for plotting.

Usage

```
prepare_volcano_input(posterior, annotation, null.effect = 0, CrI_level = 0.95)
```

Arguments

`posterior`	A data frame of posterior draws (one row per draw) `extract_fit()`.
`annotation`	A data frame with at least one column: • `parameter`: the parameter name (e.g., `doubling.1`, `logOR.treatment`) • `label`: the biological label (e.g., `cell.line`, `time.point`) • Optional: other columns (e.g., `group`, `condition`) for future coloring
`null.effect`	Central parameter value corresponding to no effect (default t=0).
`CrI_level`	a scalar between 0 and 1 specifying the mass within the credible interval (default=0.95, i.e. 95% credible interval (CrI)).

Details

Only returns pi-values and credible intervals for parameters that are **in posterior and annotation**. For formula see README or Vignette

Value

A list with:

- `result`: A data frame with columns:
  - `parameter`: original parameter name
  - `pi.value`: calculated pi.value
  - `null.effect`: set null effect by user
  - `parameter.median`: median posterior parameter value
  - `parameter.low`: lower boundary of CrI of parameter value
  - `paramter.high`: upper boundary of CrI of parameter value
  - `CrI.width`: the absolute distance between parameter.low and parameter.high
  - `CrI.level`: CrI_level set by user
  - `label`: biological label (e.g., `cell.line`)
  - Other columns from `annotation` (e.g., `group`, `condition`)

See Also

`extract_fit()`

Examples

```
# Example: Simulate posterior and annotation
posterior <- data.frame(
  doubling.1 = rnorm(1000),
  doubling.2 = rnorm(1000)
)

annotation <- data.frame(
  parameter = c("doubling.1", "doubling.2"),
  label = c("cell.line.A", "cell.line.B"),
  group = c("group1", "group1")
)

result <- prepare_volcano_input(
  posterior = posterior,
  annotation = annotation,
)

head(result$result)
```

Index