

Package ‘rstanemax’

October 14, 2022

Version 0.1.3

Title Emax Model Analysis with 'Stan'

Description Perform sigmoidal Emax model fit using 'Stan' in a formula notation, without writing 'Stan' model code.

Encoding UTF-8

License GPL-3 | file LICENSE

LazyData true

ByteCompile true

Depends R (>= 3.4.0), methods,Rcpp (>= 1.0.0)

Imports rstan (>= 2.18.2), rstantools (>= 1.5.1), magrittr (>= 1.5),
dplyr (>= 0.8.0), tidyr (>= 1.0.0), purrr (>= 0.3.0), ggplot2
(>= 2.2.1)

LinkingTo StanHeaders (>= 2.18.1), rstan (>= 2.18.2), BH (>= 1.69.0-1), Rcpp (>= 1.0.0), RcppEigen (>= 0.3.3.5.0)

SystemRequirements GNU make

NeedsCompilation yes

RoxygenNote 7.1.1

Suggests testthat, knitr, rmarkdown, spelling

VignetteBuilder knitr

Language en-US

URL <https://github.com/yoshidk6/rstanemax>

BugReports <https://github.com/yoshidk6/rstanemax/issues>

Author Kenta Yoshida [aut, cre] (<<https://orcid.org/0000-0003-4967-3831>>),
Trustees of Columbia University [cph] (src/init.cpp, tools/make_cc.R,
R/stanmodels.R)

Maintainer Kenta Yoshida <6.kurabupasu@gmail.com>

Repository CRAN

Date/Publication 2020-11-24 07:50:02 UTC

R topics documented:

rstanemax-package	2
exposure.response.sample	2
exposure.response.sample.test	3
posterior_predict	3
stanemax-methods	5
stan_emax	6

Index	8
--------------	----------

rstanemax-package	<i>The 'rstanemax' package.</i>
-------------------	---------------------------------

Description

Perform sigmoidal Emax model fit using Stan without writing Stan model code.

References

Stan Development Team (2018). RStan: the R interface to Stan. R package version 2.18.2.
<http://mc-stan.org>

exposure.response.sample	<i>Sample simulated data for exposure-response.</i>
--------------------------	---

Description

Sample simulated data for exposure-response.

Usage

```
exposure.response.sample
```

Format

A data frame with columns:

dose Dose levels used for simulation of pharmacokinetics

exposure Simulated exposure

response Simulated pharmacodynamic response

Examples

```
exposure.response.sample
```

```
exposure.response.sample.test
```

Sample simulated data for exposure-response with covariates for package testing

Description

Sample simulated data for exposure-response with covariates for package testing

Usage

```
exposure.response.sample.test
```

Format

A data frame with columns:

dose Dose levels used for simulation of pharmacokinetics

conc Simulated exposure

resp Simulated pharmacodynamic response

cov1 Covariate 1 for e0

cov2 Covariate 2 for emax

cov3 Covariate 3 for ec50 (data type factor)

cov3num Covariate 3 for ec50 (data type numeric)

Examples

```
exposure.response.sample
```

```
posterior_predict
```

Outcome prediction from posterior distribution of parameters

Description

Compute outcome predictions using posterior samples. Exposure data for prediction can be either original data used for model fit or new data.

Usage

```
## S3 method for class 'stanemax'
posterior_predict(
  object,
  newdata = NULL,
  returnType = c("matrix", "dataframe", "tibble"),
  newDatatype = c("raw", "modelframe"),
  ...
)

posterior_predict_quantile(
  object,
  newdata = NULL,
  ci = 0.9,
  pi = 0.9,
  newDatatype = c("raw", "modelframe")
)
```

Arguments

object	A stanemax class object
newdata	An optional data frame that contains columns needed for model to run (exposure and covariates). If the model does not have any covariate, this can be a numeric vector corresponding to the exposure metric.
returnType	An optional string specifying the type of return object.
newDatatype	An optional string specifying the type of newdata input, whether in the format of an original data frame or a processed model frame. Mostly used for internal purposes and users can usually leave at default.
...	Additional arguments passed to methods.
ci	Credible interval of the response without residual variability.
pi	Prediction interval of the response with residual variability.

Details

Run `vignette("emaxmodel", package = "rstanemax")` to see how you can use the posterior prediction for plotting estimated Emax curve.

Value

An object that contain predicted response with posterior distribution of parameters. The default is a matrix containing predicted response. Each row of the matrix is a vector of predictions generated using a single draw of the model parameters from the posterior distribution.

If either `dataframe` or `tibble` is specified, the function returns a data frame or tibble object in a long format - each row is a prediction generated using a single draw of the model parameters and a corresponding exposure.

Two types of predictions are generated with this function. `respHat` corresponds to the prediction without considering residual variability and is intended to provide credible interval of "mean" response. `response` include residual variability in its calculation, therefore the range represents prediction interval of observed response.

The return object also contains exposure and parameter values used for calculation.

With `posterior_predict_quantile()` function, you can obtain quantiles of `respHat` and `response` as specified by `ci` and `pi`.

stanemax-methods *Methods for stanemax objects*

Description

Methods for stanemax objects

Usage

```
## S3 method for class 'stanemax'
print(x, digits_summary = 2, ...)

extract_stanfit(x)

extract_obs_mod_frame(x)

## S3 method for class 'stanemax'
plot(x, show.ci = TRUE, show.pi = FALSE, ci = 0.9, pi = 0.9, ...)
```

Arguments

<code>x</code>	An object of class <code>stanemax</code>
<code>digits_summary</code>	The number of significant digits to use when printing the summary, defaulting to 2. Applies to the quantities other than the effective sample size, which is always rounded to the nearest integer.
<code>...</code>	Additional arguments passed to methods.
<code>show.ci</code>	An logical specifying if the output figure include credible interval of posterior prediction. Default TRUE.
<code>show.pi</code>	An logical specifying if the output figure include prediction interval. Default FALSE.
<code>ci</code>	Credible interval range.
<code>pi</code>	Prediction interval range.

 stan_emax

Bayesian Emax model fit with Stan

Description

Run sigmoidal Emax model fit with formula notation

Usage

```
stan_emax(
  formula,
  data,
  gamma.fix = 1,
  e0.fix = NULL,
  emax.fix = NULL,
  priors = NULL,
  param.cov = NULL,
  ...
)
```

Arguments

formula	a symbolic description of variables for Emax model fit.
data	an optional data frame containing the variables in the model.
gamma.fix	a (positive) numeric or NULL to specify gamma (Hill coefficient) in the sigmoidal Emax model. If NULL, gamma will be estimated from the data. If numeric, gamma is fixed at the number provided. Default = 1 (normal Emax model).
e0.fix	a numeric or NULL to specify E0 in the Emax model. If NULL, E0 will be estimated from the data. If numeric, E0 is fixed at the number provided. Default = NULL (estimate from the data).
emax.fix	a numeric or NULL to specify Emax in the Emax model. If NULL, Emax will be estimated from the data. If numeric, Emax is fixed at the number provided. Default = NULL (estimate from the data).
priors	a named list specifying priors of parameters (ec50, emax, e0, gamma, sigma). Each list item should be length 2 numeric vector, one corresponding to mean and another corresponding to standard deviation. Currently only supports normal distribution for priors.
param.cov	a named list specifying categorical covariates on parameters (ec50, emax, e0). Convert a column into factor if specific order of covariates are needed.
...	Arguments passed to <code>rstan::sampling</code> (e.g. iter, chains).

Details

The following structure is used for the Emax model:

$$Response = e_0 + e_{max} \times exposure^\gamma / (ec50^\gamma + exposure^\gamma) + \epsilon$$

$$\epsilon \sim N(0, \sigma^2)$$

Value

An object of class stanemax

Examples

```
## Not run:
data(exposure.response.sample)
fit1 <- stan_emax(response ~ exposure, data = exposure.response.sample,
  # the next line is only to make the example go fast enough
  chains = 1, iter = 500, seed = 12345)

print(fit1)

# Set priors manually, also estimate gamma instead of the default of fix to 1
fit2 <- stan_emax(response ~ exposure, data = exposure.response.sample, gamma.fix = NULL,
  priors = list(ec50 = c(100, 30), emax = c(100, 30), e0 = c(10, 5),
    gamma = c(0, 3), sigma = c(0, 30)),
  # the next line is only to make the example go fast enough
  chains = 1, iter = 500, seed = 12345)

print(fit2)

data(exposure.response.sample.test)
# Specify covariates
fit3 <- stan_emax(formula = resp ~ conc, data = exposure.response.sample.test,
  param.cov = list(emax = "cov2", ec50 = "cov3", e0 = "cov1"),
  # the next line is only to make the example go fast enough
  chains = 1, iter = 500, seed = 12345)

print(fit3)

## End(Not run)
```

Index

* datasets

- exposure.response.sample, 2
- exposure.response.sample.test, 3

- exposure.response.sample, 2
- exposure.response.sample.test, 3
- extract_obs_mod_frame
(stanemax-methods), 5
- extract_stanfit (stanemax-methods), 5

- plot.stanemax (stanemax-methods), 5
- posterior_predict, 3
- posterior_predict_quantile
(posterior_predict), 3
- posterior_predict_quantile(), 5
- print.stanemax (stanemax-methods), 5

- rstan::sampling, 6
- rstanemax (rstanemax-package), 2
- rstanemax-package, 2

- stan_emax, 6
- stanemax-methods, 5