

# Package ‘MetaStan’

November 15, 2018

**Version** 0.0.2

**Title** Bayesian Meta-Analysis via 'Stan'

**Description** Performs Bayesian meta-analysis using 'Stan'.

Includes binomial-normal hierarchical models and option to use weakly informative priors for the heterogeneity parameter and the treatment effect parameter which are described in Guenhan, Roeber, and Friede (2018) <arXiv:1809.04407>.

**License** GPL (>= 3)

**Encoding** UTF-8

**Date** 2018-11-14

**LazyData** true

**ByteCompile** true

**Depends** R (>= 3.4.0), Rcpp (>= 0.12.17), methods

**Imports** rstan (>= 2.18.1), rstantools (>= 1.5.0)

**Suggests** testthat, knitr, rmarkdown, ggplot2

**LinkingTo** StanHeaders (>= 2.18.0), rstan (>= 2.18.1), BH (>= 1.66.0-1), Rcpp (>= 0.12.17), RcppEigen (>= 0.3.3.4.0)

**SystemRequirements** GNU make

**NeedsCompilation** yes

**RoxygenNote** 6.1.0

**URL** <http://github.com/gunhanb/MetaStan>

**BugReports** <http://github.com/gunhanb/MetaStan/issues>

**VignetteBuilder** knitr

**Author** Burak Kuersad Guenhan [aut, cre],  
Trustees of Columbia University [cph] (src/init.cpp, tools/make\_cpp.R,  
R/stanmodels.R)

**Maintainer** Burak Kuersad Guenhan <burak.gunhan@med.uni-goettingen.de>

**Repository** CRAN

**Date/Publication** 2018-11-15 08:20:15 UTC

## R topics documented:

MetaStan-package . . . . .	2
convert_data_arm . . . . .	3
dat.Berkey1995 . . . . .	4
dat.Crins2014 . . . . .	4
meta_stan . . . . .	5
print.meta_stan . . . . .	7
<b>Index</b>	<b>9</b>

---

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

---

### Description

Fitting Bayesian meta-analysis models via Rstan.

### Details

To fit meta-analysis models using frequentist methods, there are many R packages available including 'metafor'. On the other hand, Bayesian estimation methods such as Markov chain Monte Carlo (MCMC) are very attractive for meta-analysis, especially because they can be used to fit more complicated models. These include binomial-normal hierarchical models and beta-binomial models which are based on the exact distributional assumptions unlike (commonly used) normal-normal hierarchical model. Another advantage of Bayesian methods to be able to use informative prior distributions for example to regularize heterogeneity estimates in case of low number of studies. Thus, we developed 'MetaStan' which uses Stan (a modern MCMC engine) to fit several pairwise meta-analysis models including binomial-normal hierarchical model and beta-binomial model. This package is also the accompanying package of Guenhan et al (2018).

### Author(s)

Burak Kuersad Guenhan <burak.gunhan@med.uni-goettingen.de>

### References

- Stan Development Team (2018). RStan: the R interface to Stan. R package version 2.17.3. <http://mc-stan.org>
- Günhan, B and Röver, C and Friede, T (2018). Meta-analysis of few studies involving rare events. arXiv.org E-print Archive. <https://arxiv.org/abs/1809.04407>.

---

convert_data_arm	<i>Convert contrast-based dataset to arm-based dataset</i>
------------------	--

---

## Description

convert\_data\_arm creates a dataframe to fit a Binomial-Normal Hierarchical model using glmer function.

## Usage

```
convert_data_arm(nt, nc, pt, pc)
```

## Arguments

nt	Number of subjects in treatment arm
nc	Number of subjects in control arm
pt	Number of events in treatment arm
pc	Number of events in control arm

## Value

A dataframe object

## Examples

```
data('dat.Crins2014', package = "MetaStan")
## Subset of dataset where PTLD outcomes available
dat.Crins2014.PTLD = subset(dat.Crins2014, is.finite(exp.PTLD.events))
## Create arm-based dataset
dat.Crins2014.PTLD.arm <- convert_data_arm(dat.Crins2014.PTLD$exp.total,
dat.Crins2014.PTLD$cont.total, dat.Crins2014.PTLD$exp.PTLD.events,
dat.Crins2014.PTLD$cont.PTLD.events)

glmer(cbind(r, sampleSize - r) ~ factor(mu) + factor(theta) + (theta12 - 1|mu),
data = dat.Crins2014.PTLD.long, family = binomial(link = "logit"), nAGQ = 7)
```

---

dat.Berkey1995

*Trials investigating effectiveness of the BCG vaccine against TB*

---

### Description

A dataset containing the results from 13 trials examining the efficacy of Bacillus Calmette-Guerin (BCG) vaccine against tuberculosis (TB).

### Usage

dat.Berkey1995

### Format

A data frame with following columns

**Trial** Trial number

**rt** number of TB events in treatment arm

**nt** number of subjects in treatment arm

**rc** number of TB events in control arm

**nc** number of subjects in control arm

**Latitude** absolute latitude of the study location

### Source

Berkey, C.S., Hoaglin, D.C., Mosteller, F. and Colditz, G.A., 1995. A random-effects regression model for meta-analysis. *Statistics in medicine*, 14(4), pp.395-411

---

dat.Crins2014

*Pediatric liver transplant example data*

---

### Description

Numbers of cases and events (PTLDs or deaths) in experimental and control groups of six studies.

### Usage

dat.Crins2014

**Format**

A data frame with following columns

**publication** publication identifier (first author and publication year)

**year** publication year

**randomized** randomization status (y/n)

**control.type** type of control group ("concurrent" or "historical")

**comparison** type of comparison ("IL-2RA only", "delayed CNI", or "no/low steroids")

**followup** t follow-up time in months

**exp.PTLD.events** number of PTLD events in experimental group

**exp.death.events** number of deaths in experimental group

**exp.total** number of patients in experimental group

**cont.PTLD.events** number of PTLD events in control group

**cont.death.events** number of deaths in control group

**cont.total** number of patients in control group

**Source**

N.D. Crins, C. Roever, A.D. Goralczyk, T. Friede. Interleukin-2 receptor antagonists for pediatric liver transplant recipients: A systematic review and meta-analysis of controlled studies. *Pediatric Transplantation*, 18(8):839-850, 2014.

---

meta\_stan

*Fitting a meta-analysis model using Stan*

---

**Description**

'meta\_stan' fits a meta-analysis model using Stan.

**Usage**

```
meta_stan(ntrt = NULL, nctrl = NULL, rtrt = NULL, rctrl = NULL,
  model = "BNHM1", mu_prior = c(0, 10), theta_prior = NULL,
  tau_prior = 0.5, tau_prior_dist = "half-normal", delta = NULL,
  chains = 4, iter = 2000, warmup = 1000, adapt_delta = 0.95)
```

**Arguments**

ntrt	Number of subjects in treatment arm
nctrl	Number of subjects in control arm
rtrt	Number of events in treatment arm
rctrl	Number of events in control arm

model	A string specifying the model used. Available options are ‘FE‘ (fixed-effect model using binomial likelihood) ‘BNHM1‘ (Model 4 from <i>Jackson et al (2018)</i> ), ‘BNHM2‘ (Model 2 from <i>Jackson et al (2018)</i> ), and ‘Beta-binomial‘ (Beta-binomial model from <i>Kuss (2014)</i> ). Default is ‘BNHM1‘.
mu_prior	A numerical vector specifying the parameter of the normal prior density for baseline risks, first value is parameter for mean, second is for variance. Default is $c(0, 10)$ .
theta_prior	A numerical vector specifying the parameter of the normal prior density for treatment effect estimate, first value is parameter for mean, second is for variance. Default is NULL.
tau_prior	A numerical value specifying the standard dev. of the prior density for heterogeneity stdev. Default is 0.5.
tau_prior_dist	A string specifying the prior density for the heterogeneity standard deviation, option is ‘half-normal‘ for half-normal prior, ‘uniform‘ for uniform prior, ‘half-cauchy‘ for half-cauchy prior.
delta	A numerical value specifying the upper bound of the a priori interval for treatment effect on odds ratio scale <i>Guenhan et al (2018)</i> . This is used to calculate a normal weakly informative prior for theta. Thus when this argument is specified, ‘theta‘ should be left empty. Default is NULL.
chains	A positive integer specifying the number of Markov chains. The default is 4.
iter	A positive integer specifying the number of iterations for each chain (including warmup). The default is 2000.
warmup	A positive integer specifying the number of warmup (aka burnin) iterations per chain. The default is 1000.
adapt_delta	A numerical value specifying the target average proposal acceptance probability for adaptation. See Stan manual for details. Default is 0.95. In general you should not need to change adapt_delta unless you see a warning message about divergent transitions, in which case you can increase adapt_delta from the default to a value closer to 1 (e.g. from 0.95 to 0.99, or from 0.99 to 0.999, etc).

## Value

an object of class ‘stanfit‘ returned by ‘rstan::sampling‘

## References

- Guenhan BK, Roever C, Friede T. Meta-analysis of few studies involving rare events *arXiv preprint* 2018;<https://arxiv.org/abs/1809.04407>. *Stat Med* 2018;37:1059–1085.
- Jackson D, Law M, Stijnen T, Viechtbauer W, White IR. A comparison of 7 random-effects models for meta-analyses that estimate the summary odds ratio. *Stat Med* 2018;37:1059–1085.
- Kuss O, Hoyer A and Solms A. Meta-analysis for diagnostic accuracy studies: a new statistical model using beta-binomial distributions and bivariate copulas. *Stat Med*, 2014; 33:17–30. doi:10.1002/sim.5909

## Examples

```

data('dat.Crins2014', package = "MetaStan")
## Subset of dataset where PTLD outcomes available
dat.Crins2014.PTLD = subset(dat.Crins2014, is.finite(exp.PTLD.events))
## Fitting a Binomial-Normal Hierarchical model using vague priors for theta
bnhm.vague.PTLD.stan <- meta_stan(ntrt = dat.Crins2014.PTLD$exp.total,
                                nctrl = dat.Crins2014.PTLD$cont.total,
                                rtrt = dat.Crins2014.PTLD$exp.PTLD.events,
                                rctrl = dat.Crins2014.PTLD$cont.PTLD.event,
                                mu_prior = c(0, 10), theta_prior = c(0, 100),
                                tau_prior = 0.5)

## Obtaining a small summary
print(bnhm.vague.PTLD.stan)
## Extract the rstan fit for post-processing, eg convergence diagnostics
bnhm.vague.PTLD.stanfit = bnhm.vague.PTLD.stan$fit
  ## see `?rstan::sampling` for for post-processing of the `stanfit` object
## Fitting a Binomial-Normal Hierarchical model using WIP for theta
bnhm.wip.PTLD.stan <- meta_stan(ntrt = dat.Crins2014.PTLD$exp.total,
                                nctrl = dat.Crins2014.PTLD$cont.total,
                                rtrt = dat.Crins2014.PTLD$exp.PTLD.events,
                                rctrl = dat.Crins2014.PTLD$cont.PTLD.event,
                                mu_prior = c(0, 10),
                                theta_prior = c(0, 2.82),
                                tau_prior = 0.5)

## Fitting a fixed-effect Binomial model using vague priors for theta
bm.vague.PTLD.stan <- meta_stan(ntrt = dat.Crins2014.PTLD$exp.total,
                                nctrl = dat.Crins2014.PTLD$cont.total,
                                rtrt = dat.Crins2014.PTLD$exp.PTLD.events,
                                rctrl = dat.Crins2014.PTLD$cont.PTLD.event,
                                mu_prior = c(0, 10), theta_prior = c(0, 100),
                                model = "FE")

## Fitting a Beta-binomial model using vague priors
bnhm.wip.PTLD.stan <- meta_stan(ntrt = dat.Crins2014.PTLD$exp.total,
                                nctrl = dat.Crins2014.PTLD$cont.total,
                                rtrt = dat.Crins2014.PTLD$exp.PTLD.events,
                                rctrl = dat.Crins2014.PTLD$cont.PTLD.event,
                                model = "Beta-binomial")

```

---

```
print.meta_stan
```

```
Print meta_stan object
```

---

## Description

Takes an `meta_stan` object which is obtained by function `meta_stan` and print the model and data information such as model type used in the model.

**Usage**

```
## S3 method for class 'meta_stan'  
print(x, digits = 2, ...)
```

**Arguments**

<code>x</code>	A <code>meta_stan</code> object.
<code>digits</code>	An integer indicating the number of decimal places.
<code>...</code>	Further arguments passed to or from other methods.

**Details**

The resulting `data.frame` can be used as `data` argument in `meta_stan`.

**Value**

The return value is invisible `NULL`.



# Index

## \*Topic **datasets**

dat.Berkey1995, [4](#)

dat.Crins2014, [4](#)

convert\_data\_arm, [3](#)

dat.Berkey1995, [4](#)

dat.Crins2014, [4](#)

meta\_stan, [5](#)

MetaStan (MetaStan-package), [2](#)

MetaStan-package, [2](#)

print.meta\_stan, [7](#)