

Package ‘marginme’

July 28, 2025

Type Package

Title Estimation of Relative Risks, Risk Differences, and Marginal Effects from Mixed Models Using Marginal Standardization

Version 0.1.0

Author Sam Watson [aut, cre]

Description Functionality to estimate relative risks, risk differences, and partial effects from mixed model. Marginalisation over random effect terms is accomplished using Markov Chain Monte Carlo.

Imports methods, glmmrBase (>= 1.0.2)

Suggests glmmTMB (>= 1.1.10)

License GPL (>= 2)

Encoding UTF-8

LazyData true

URL <https://github.com/samuel-watson/marginme>

BugReports <https://github.com/samuel-watson/marginme/issues>

RoxygenNote 7.3.2

Depends R (>= 2.10)

NeedsCompilation no

Maintainer Sam Watson <S.I.Watson@bham.ac.uk>

Repository CRAN

Date/Publication 2025-07-28 18:50:07 UTC

Contents

confint.margin	2
margin	3
print.margin	5
summary.margin	6
trial_data	6

Index	7
--------------	----------

confint.margin	<i>Confidence interval for marginal effect</i>
----------------	--

Description

Confidence interval method for class "'margin'"

Usage

```
## S3 method for class 'margin'
confint(object, parm, level = 0.95, ...)
```

Arguments

object	An object of class "'margin'" resulting from a call to margin
parm	No effect.
level	the confidence level required
...	additional argument(s) for methods

Details

Computes confidence intervals using a standard Wald test for the marginal effect. If argument 'df' is used, then a t-statistic is use to construct the interval with 'df' degrees of freedom, otherwise a z-statistic is used.

Value

A named vector giving lower and upper confidence limits for the marginal effect. They will be labelled as (1-level)/2 and 1-(1-level).2 An argument df can be provided to use a t-test to construct the confidence interval.

Examples

```
## fit a model using glmmTMB
fit <- glmmTMB::glmmTMB(y ~ Treatment + x1 + x2 + x3 + x4 + (1|Cluster),
  data = trial_data, family = binomial(link="logit"),REML = TRUE)
## relative risk, average over random effects and fixed effects
m1 <- margin(fit,
  x = "Treatment",
  type = "ratio",
  average = c("x1", "x2", "x3", "x4"),
  re = "average",
  se="GLS")
confint(m1)
```

margin	<i>Estimates Relative Risks, Risk Differences, and Marginal Effects from Mixed Models</i>
--------	---

Description

Calculates the marginal effect of variable x . There are several options for marginal effect and several types of conditioning or averaging. The type of marginal effect can be the derivative of the mean with respect to x ('dydx'), the expected difference $E(y|x=a)-E(y|x=b)$ ('diff'), or the expected log ratio $\log(E(y|x=a)/E(y|x=b))$ ('ratio'). Other fixed effect variables can be set at specific values ('at'), set at their mean values ('atmeans'), or averaged over ('average'). Averaging over a fixed effects variable here means using all observed values of the variable in the relevant calculation. The random effects can similarly be set at their estimated value ('re="estimated"'), set to zero ('re="zero"'), set to a specific value ('re="at"'), or averaged over ('re="average"'). The standard errors are calculated using the delta method with one of several options for the variance matrix of the fixed effect parameters. Several of the arguments require the names of the variables as given to the model object. Most variables are as specified in the formula, factor variables are specified as the name of the 'variable_value', e.g. 't_1'. To see the names of the stored parameters and data variables see the member function 'names()'.

Usage

```
margin(
  fit,
  x,
  type,
  re,
  se,
  at = c(),
  atmeans = c(),
  average = c(),
  xvals = c(1, 0),
  atvals = c(),
  revals = c(),
  oim = FALSE,
  sampling = 250
)
```

Arguments

fit	Either a lme4, glmmTMB, or glmmrBase model fit.
x	String. Name of the variable to calculate the marginal effect for.
type	String. Either 'dydx' for derivative, 'diff' for difference, or 'ratio' for log ratio. See description.
re	String. Either 'estimated' to condition on estimated values, 'zero' to set to zero, 'at' to provide specific values, or 'average' to average over the random effects.

se	String. Type of standard error to use, either 'GLS' for the GLS standard errors, 'KR' for Kenward-Roger estimated standard errors, or 'KR2' for the improved Kenward-Roger correction.
at	Optional. A vector of strings naming the fixed effects for which a specified value is given.
atmeans	Optional. A vector of strings naming the fixed effects that will be set at their mean value.
average	Optional. A vector of strings naming the fixed effects which will be averaged over.
xvals	Optional. A vector specifying the values of 'a' and 'b' for 'diff' and 'ratio'. The default is (1,0).
atvals	Optional. A vector specifying the values of fixed effects specified in 'at' (in the same order).
revals	Optional. If 're="at"' then this argument provides a vector of values for the random effects.
oim	Logical. If TRUE use the observed information matrix, otherwise use the expected information matrix for standard error and related calculations.
sampling	Integer. Number of MCMC samples to use.

Value

A named vector with elements 'margin' specifying the point estimate and 'se' giving the standard error.

Examples

```
## fit a model using glmmTMB
fit <- glmmTMB::glmmTMB(y ~ Treatment + x1 + x2 + x3 + x4 + (1|Cluster),
  data = trial_data, family = binomial(link="logit"),REML = TRUE)
## relative risk, average over random effects and fixed effects
m1 <- margin(fit,
  x = "Treatment",
  type = "ratio",
  average = c("x1","x2","x3","x4"),
  re = "average",
  se="GLS")
summary(m1)
## stata default for margins command is to set random effects to zero
m2 <- margin(fit,
  x = "Treatment",
  type = "ratio",
  average = c("x1","x2","x3","x4"),
  re = "zero",
  se="GLS")
summary(m2)
## finally estimate a risk difference, with random effects at zero and fixed effects
## at mean values
m3 <- margin(fit,
```

```

      x = "Treatment",
      type = "diff",
      atmeans = c("x1", "x2", "x3", "x4"),
      re = "zero",
      se="GLS")
summary(m3)

```

print.margin	<i>Prints the marginal output</i>
--------------	-----------------------------------

Description

Print method for class "margin"

Usage

```

## S3 method for class 'margin'
print(x, ...)

```

Arguments

x	An object of class "margin" resulting from a call to margin
...	Further arguments passed from other methods

Value

No return, called for effects

Examples

```

## fit a model using glmmTMB
fit <- glmmTMB::glmmTMB(y ~ Treatment + x1 + x2 + x3 + x4 + (1|Cluster),
  data = trial_data, family = binomial(link="logit"), REML = TRUE)
## relative risk, average over random effects and fixed effects
margin(fit,
  x = "Treatment",
  type = "ratio",
  average = c("x1", "x2", "x3", "x4"),
  re = "average",
  se="GLS")

```

summary.margin	<i>Summarises the marginal output</i>
----------------	---------------------------------------

Description

Summary method for "'margin'" class

Usage

```
## S3 method for class 'margin'
summary(object, ...)
```

Arguments

object	An object of class "'margin'" resulting from a call to margin
...	Further arguments passed from other methods

Value

No return, called for effects

Examples

```
## fit a model using glmmTMB
fit <- glmmTMB::glmmTMB(y ~ Treatment + x1 + x2 + x3 + x4 + (1|Cluster),
  data = trial_data, family = binomial(link="logit"), REML = TRUE)
## relative risk, average over random effects and fixed effects
m1 <- margin(fit,
  x = "Treatment",
  type = "ratio",
  average = c("x1", "x2", "x3", "x4"),
  re = "average",
  se="GLS")
summary(m1)
```

trial_data	<i>Simulated trial data</i>
------------	-----------------------------

Description

Simulated trial data used to demonstrate the estimation of relative risk from an adjusted mixed logistic regression model. See [margin](#).

Index

`confint.margin`, [2](#)

`margin`, [3](#), [6](#)

`print.margin`, [5](#)

`summary.margin`, [6](#)

`trial_data`, [6](#)