Package ‘glmmsr’

February 4, 2019

Title  Fit a Generalized Linear Mixed Model

Version  0.2.3

Description  Conduct inference about generalized linear mixed models, with a choice about which method to use to approximate the likelihood. In addition to the Laplace and adaptive Gaussian quadrature approximations, which are borrowed from 'lme4', the likelihood may be approximated by the sequential reduction approximation, or an importance sampling approximation. These methods provide an accurate approximation to the likelihood in some situations where it is not possible to use adaptive Gaussian quadrature.

Depends  R (>= 3.2.0)

LinkingTo  Rcpp, RcppEigen, BH

Imports  lme4 (>= 1.1-8), Matrix, R6, Rcpp, methods, stats, utils, numDeriv

URL  http://github.com/heogden/glmmsr

BugReports  http://github.com/heogden/glmmsr/issues

License  GPL (>= 2)

LazyData  true

Suggests  BradleyTerry2, knitr, mdhglm, rmarkdown, testthat

VignetteBuilder  knitr

RoxygenNote  6.1.1

NeedsCompilation  yes

Author  Helen Ogden [aut, cre]

Maintainer  Helen Ogden <heogden12@gmail.com>

Repository  CRAN

Date/Publication  2019-02-04 11:09:22 UTC
find_lfun_glmm

Find the log-likelihood function

Description

Find the log-likelihood function

Usage

find_lfun_glmm(modfr, method, control = NULL, lme4_control = set_lme4_control())

Arguments

modfr a model frame, the output of find_modfr_glmm

method the method used to approximate the likelihood. The options are "Laplace", "AGQ" (the adaptive Gaussian quadrature approximation, from lme4), "SR" (the sequential reduction approximation) and "IS" (an importance sampling approximation).

control a list of extra parameters controlling the approximation to the likelihood. See 'Details' for more information.

lme4_control the result of a call to lme4_control, containing control parameters passed to lme4. See ?lme4_control.
find_modfr_glmm  Parse a formula (and possibly subformulas)

Description
Parse a formula (and possibly subformulas)

Usage
find_modfr_glmm(formula, subformula = NULL, data = NULL,
family = gaussian, weights = NULL, offset = NULL,
lme4_control = set_lme4_control())

Arguments
formula  a two-sided linear formula object describing both the fixed-effects and random-effects part of the model, with the response on the left of a \( \sim \) operator and the terms, separated by + operators, on the right. Random-effects terms are distinguished by vertical bars ("|") separating expressions for design matrices from grouping factors.
subformula  a subformula, describing how a substituted variable depends on covariates, or a list of subformulas, if there is more than one Sub() term in formula.
data  an optional data frame, list or environment containing the variables named in formula, and in any of the subformulas.
family  a GLM family, see glm and family.
weights  an optional vector of 'prior weights' to be used in the fitting process. Should be NULL or a numeric vector.
offset  this can be used to specify an a priori known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases. One or more offset terms can be included in the formula instead or as well, and if more than one is specified their sum is used. See model.offset.
lme4_control  the result of a call to lme4_control, containing control parameters passed to lme4. See ?lme4_control.

glm  Fit a GLMM

Description
Fit a GLMM
Usage

glmm(formula, subformula = NULL, data = NULL, family = gaussian, method = NULL, control = list(), weights = NULL, offset = NULL, prev_fit = NULL, verbose = 1L, lme4_control = set_lme4_control())

Arguments

formula a two-sided linear formula object describing both the fixed-effects and random-effects part of the model, with the response on the left of a ~ operator and the terms, separated by + operators, on the right. Random-effects terms are distinguished by vertical bars ("|") separating expressions for design matrices from grouping factors.

subformula a subformula, describing how a substituted variable depends on covariates, or a list of subformulas, if there is more than one Sub() term in formula.

data an optional data frame, list or environment containing the variables named in formula, and in any of the subformulas.

family a GLM family, see glm and family.

method the method used to approximate the likelihood. The options are "Laplace", "AGQ" (the adaptive Gaussian quadrature approximation, from lme4), "SR" (the sequential reduction approximation) and "IS" (an importance sampling approximation).

control a list of extra parameters controlling the approximation to the likelihood. See 'Details' for more information.

weights an optional vector of ‘prior weights’ to be used in the fitting process. Should be NULL or a numeric vector.

offset this can be used to specify an a priori known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases. One or more offset terms can be included in the formula instead or as well, and if more than one is specified their sum is used. See model.offset.

prev_fit a glmmFit object, the result of a previous model fit.

verbose controls how much detail to print out while fitting the model. For verbose = 0, print nothing. For verbose = 1 (the default), print output approximately once a second during model fitting. For verbose = 2, print out the parameter value and log-likelihood at every stage of optimization.

lme4_control the result of a call to lme4_control, containing control parameters passed to lme4. See ?lme4_control.

Details

The control argument is a list, used to specify further arguments controlling the approximation to the likelihood:

nAGQ the number of adaptive Gaussian quadrature points. Only used if method = "AGQ". Defaults to 15.
nSL the level of sparse grid storage. Only used if method = "SR". Defaults to 3.

nIS the number of samples to use for importance sampling. Only used if method = "IS". Defaults to 1000.

order the order of Laplace approximation. only used if method = "Laplace". Defaults to 1.

check_Laplace should quality of first-order Laplace approximation be checked? Only used if method = "Laplace" and order = 1. Defaults to TRUE.

divergence_threshold if check_Laplace = TRUE, warn about quality of inference using the first-order Laplace approximation if measure of divergence from inference with second-order Laplace approximation exceeds divergence_threshold. Defaults to 0.1.

Value

An object of the class glmmFit

Examples

# Fit a three-level model with the Laplace approximation to the likelihood
(mod_Laplace <- glmm(response ~ covariate + (1 | cluster) + (1 | group),
                     data = three_level, family = binomial,
                     method = "Laplace"))

# if we try to fit with adaptive Gaussian quadrature, we get an error
## Not run:
(mod_AGQ <- glmm(response ~ covariate + (1 | cluster) + (1 | group),
                  data = three_level, family = binomial, method = "AGQ",
                  control = list(nAGQ = 15)))

## End(Not run)

# We can fit with the Sequential Reduction approximation
## Not run:
(mod_SR <- glmm(response ~ covariate + (1 | cluster) + (1 | group),
                 data = three_level, family = binomial, method = "SR",
                 control = list(nSL = 3)))

## End(Not run)

# the estimates of the random effects standard deviations
# are larger than those using the Laplace approximation

Description

The glmmsr package provides functions to conduct inference about generalized linear mixed models, giving the user a choice about which method to use to approximate the likelihood.
Details

In addition to the Laplace and adaptive Gaussian quadrature approximations, which are borrowed
from lme4, the likelihood may be approximated by the sequential reduction approximation or an
importance sampling approximation. These methods provide an accurate approximation to the
likelihood in some situations where it is not possible to use adaptive Gaussian quadrature.

The main function of the glmmsr package is glmm, which is used to fit the GLMM. Its interface al-

lows a larger class of models than those allowed by lme4, including structured pairwise comparison
models.

References


set_lme4_control

Control of Mixed Model Fitting

Description

A version of glmerControl from lme4, with different defaults.

Usage

set_lme4_control(check.nobs.vs.rankZ = "ignore",
check.nobs.vs.nlev = "ignore", check.nlev.gtreq.5 = "ignore",
check.nlev.gtr.1 = "ignore", check.nobs.vs.nRE = "ignore",
check.rankX = c("message+drop.cols", "silent.drop.cols",
"warn+drop.cols", "stop.deficient", "ignore"),
check.scaleX = "warning", check.formula.LHS = "stop",
check.response.not.const = "ignore", ...)

Arguments

check.nobs.vs.rankZ
character - rules for checking whether the number of observations is greater than
(or greater than or equal to) the rank of the random effects design matrix (Z),
usually necessary for identifiable variances. As for action, with the addition of
"warningSmall" and "stopSmall", which run the test only if the dimensions
of Z are < 1e6. nobs > rank(Z) will be tested for LMMs and GLMMs with
estimated scale parameters; nobs >= rank(Z) will be tested for GLMMs with
fixed scale parameter. The rank test is done using the method="qr" option of
the rankMatrix function.

check.nobs.vs.nlev
character - rules for checking whether the number of observations is less than
(or less than or equal to) the number of levels of every grouping factor, usually
necessary for identifiable variances. As for action, nobs<=nlevels will be
tested for LMMs and GLMMs with estimated scale parameters; nobs<=nlevels
will be tested for GLMMs with fixed scale parameter.
three_level

A dataset simulated from a three-level model

Description

A dataset simulated from a three-level model

Usage

three_level

Format

An object of class list of length 4.
Examples

# Fit a three-level model with the Laplace approximation to the likelihood
(mod_Laplace <- glmm(response ~ covariate + (1 | cluster) + (1 | group),
data = three_level, family = binomial,
method = "Laplace")

# if we try to fit with adaptive Gaussian quadrature, we get an error
## Not run:
(mod_AGQ <- glmm(response ~ covariate + (1 | cluster) + (1 | group),
data = three_level, family = binomial, method = "AGQ",
control = list(nAGQ = 15)))

## End(Not run)

# We can fit with the Sequential Reduction approximation
## Not run:
(mod_SR <- glmm(response ~ covariate + (1 | cluster) + (1 | group),
data = three_level, family = binomial, method = "SR",
control = list(nSL = 3)))

## End(Not run)

# the estimates of the random effects standard deviations
# are larger than those using the Laplace approximation

---

two_level  

A dataset simulated from a two-level model

Description

A dataset simulated from a two-level model

Usage

two_level

Format

An object of class list of length 3.

Examples

# Fit a two-level model with the Laplace approximation to the likelihood
(mod_Laplace <- glmm(response ~ covariate + (1 | cluster), data = two_level,
family = binomial, method = "Laplace")

# or with adaptive Gaussian quadrature
(mod_AGQ <- glmm(response ~ covariate + (1 | cluster), data = two_level,
family = binomial, method = "AGQ", control = list(nAGQ = 15)))
# or with the Sequential Reduction approximation
(mod_SR <- glm(response ~ covariate + (1 | cluster), data = two_level,
               family = binomial, method = "SR", control = list(nSL = 3)))

# in a two-level model, method = "SR" is equivalent to method = "AGQ" with
# nAGQ = 2^(nSL+1) - 1
Index

*Topic datasets
  three_level, 7
  two_level, 8

family, 3, 4
find_lfun_glmm, 2
find_modfr_glmm, 3

glm, 3, 4
glmm, 3, 6
glmmr, 5
glmmr-package (glmmr), 5

model.offset, 3, 4

offset, 3, 4

rankMatrix, 6, 7

set_lme4_control, 6

three_level, 7
two_level, 8