Package 'trouBBlme4SolveR'

August 8, 2022

Version 0.1.1

Date 2022-08-08

Title Troubles Solver for 'lme4'

Description The main function of the package aims to update 'lmer()'/'glmer()' models depending on their warnings, so trying to avoid convergence and singularity problems.

Depends R (>= 4.0)

Imports lme4 (>= 1.1-28)

Suggests nlme (>= 3.1-123), ggplot2 (>= 2.0.0), minqa

License GPL-3

URL https://gitlab.com/iagogv/trouBBlme4SolveR

BugReports https://gitlab.com/iagogv/trouBBlme4SolveR/-/issues

Encoding UTF-8

NeedsCompilation no

Author Iago Giné-Vázquez [aut, cre] (<https://orcid.org/0000-0002-6725-2638>)

Maintainer Iago Giné-Vázquez <iago.gin-vaz@protonmail.com>

Repository CRAN

Date/Publication 2022-08-08 16:30:02 UTC

R topics documented:

---

dwmw   *Solver for (generalized) linear MM warnings*

Description

Updates a (possibly generalized) linear mixed model computed with the library **lme4** functions
`lmer` or `glmer`, in order to elude some singularity or convergence problems informed by distinct
warnings or messages.

Usage

```
dwmw(lmmodel,
     boundary_check = TRUE, scale = FALSE, scale_info = TRUE,
     tol = 1e-4, max_message_iter = 7, max_nAGQ = 6,
     next_optimizer = "bobyqa", next_optCtrl = list(maxfun = 2e5),
     when_next = max_message_iter - 1, verbose = FALSE)
```

Arguments

`lmmodel`	The model of interest, output of either `lmer` or `glmer`, that is, an object of class `merMod` and *subclass* either lmerMod or glmerMod.
`boundary_check`	A length 1 boolean object (`TRUE` [default] or `FALSE`), determining if the function removes those random effects which make the model singular (in the simplest situation, those with standard deviation almost 0.
`scale`	A length 1 boolean object (`TRUE` or `FALSE` [default]), determining if the function must rescale the numeric predictors when this action is asked in a *warning* or *message*.
`scale_info`	A length 1 boolean object (`TRUE` [default] or `FALSE`), determining if the function should cat a message when some predictor variables are on very different scales, but are not rescaled.
`tol`	A numeric value (default is `1e-4`), tolerance under which to consider if a model is singular (see boundary_check argument).
`max_message_iter`	
	A numeric value (default is 7), generally the maximum number of iterations allowed to update the model. For the exception, see the Details section.
`max_nAGQ`	A numeric value (default is 6), maximum to which increase the nAGQ argument fitting `glmer` models.
`next_optimizer`	An optimizer (default is `"bobyqa"`), next to be used when model convergence is failing.
`next_optCtrl`	A `list` (default is `list(maxfun=2e5)`) of additional arguments to be passed to the nonlinear optimizer (see `Nelder_Mead`, `bobyqa`). In particular, both Nelder_Mead and bobyqa use maxfun to specify the maximum number of function evaluations they will try before giving up - in contrast to `optim` and optimx-wrapped optimizers, which use maxit. (Also see `convergence` for details of stopping tolerances for different optimizers.

when_next
: A numeric value not greater than max_message_iter - 1 (which is the default), specifying from which iteration a model convergence failing is approached with a new optimizer (next_optimizer) and with a new set of arguments to it (next_optCtrl).

verbose
: A length 1 boolean object (TRUE or FALSE [default])specifying if function iterations and model updating should be verbose.

Details

If the model does not generate any warning or message, dwmw returns itself. The function rescales numeric predictors if some warning or message suggests it and the scale argument is TRUE. If the model fails to converge, it only updates the starting values for the parameters in the model through the first when_next - 1 iterations, while it also updates the optimizer and their arguments through the next iterations.

If the model is singular (there is a *"boundary (singular)"*-alike message) and boundary_check is TRUE, the formula is updated removing the random effects which cause the model to be singular. If all the random effects are removed, a linear or generalized linear model is computed according to the lmmodel object.

If the model has subclass glmerMod, a single scalar random effect and converges but it is nearly unidentifiable with a very large eigenvalue, then nAGQ is increased one unity while nAGQ <= max_nAGQ (and only in this case, allowing the iterations to exceed max_message_iter until nAGQ == max_nAGQ).

Value

Generally, an object of class [merMod](), being either an object of *subclass* glmerMod or an object of subclass lmerMod, corresponding to the input lmmodel argument. When all random effects are removed, an object of class glm or lm depending also on the input lmmodel.

Author(s)

Iago Giné-Vázquez, <iago.gin-vaz@protonmail.com>

See Also

[lmer](), [glmer](), [merMod]() [lmerControl](), [convergence](), [troubleshooting](), [isSingular](), [lm]() and [glm]().

Examples

```
if(requireNamespace("nlme")){
library(lme4)
data(Orthodont, package = "nlme")
Orthodont$nsex <- as.numeric(Orthodont$Sex == "Male")
Orthodont$nsexage <- with(Orthodont, nsex*age)
## Without using dwmw, the next model is singular
fmo <- lmer(distance ~ age + (age|Subject) + (0+nsex|Subject) +
    (0 + nsexage|Subject), data = Orthodont)
summary(fmo)

## Using dwmw, we remove the singularity of the model
fmo1 <- dwmw(lmer(distance ~ age + (age|Subject) + (0+nsex|Subject) +
```

```
  (0 + nsexage|Subject), data = Orthodont))
summary(fmo1)

## It also can be used on the output object

fmo2 <- dwmw(fmo)
summary(fmo2)
}
```

---

fly_parameters	*Data for the SO question* R*hrefhttps://stackoverflow.com/questions/23478792/warning-messages-when-trying-to-run-glmer-in-rwarning messages when trying to run glmer in r.*

Description

Two integer variables (ValidDetections and FalseDetections) to build an outcome, two factor variables to be used as clusters for the random effects (SUR.ID and Day), three factor variables to be used as fixed effects (tm, Area and replicate), five numeric variables to be used as fixed effects (c.distance, c.tm.depth, c.receiver.depth, c.temp and c.wind) and a pair of variables extra, allowing to build a model which fails to converge.

Usage

```
data("fly_parameters")
```

Format

A data frame with 220 observations on the following 14 variables.

SUR.ID a factor with levels 10185 10186 10250

tm a factor with levels CT PT-04

ValidDetections a numeric vector

CountDetections a numeric vector

FalseDetections a numeric vector

replicate a factor with levels 1 2

Area a factor with levels Drug Channel Finger

Day a factor with levels 03/06/13 2/22/13 2/26/13 2/27/13 3/14/13

R.det a numeric vector

c.receiver.depth a numeric vector

c.tm.depth a numeric vector

c.temp a numeric vector

c.wind a numeric vector

c.distance a numeric vector

Source

warning messages when trying to run glmer in r

Examples

```
data(fly_parameters)
str(fly_parameters)
df <- fly_parameters
df$SUR.ID <- factor(df$SUR.ID)
df$replicate <- factor(df$replicate)
Rdet <- cbind(df$ValidDetections,df$FalseDetections)
Unit <- factor(1:length(df$ValidDetections))

library(lme4)
m1 <- glmer(Rdet ~ tm:Area + tm:c.distance +
            c.distance:Area + c.tm.depth:Area +
            c.receiver.depth:Area + c.temp:Area +
            c.wind:Area +
            c.tm.depth + c.receiver.depth +
            c.temp +c.wind + tm + c.distance + Area +
            replicate +
            (1|SUR.ID) + (1|Day) + (1|Unit) ,
            data = df, family = binomial(link="logit"))
summary(m1)
m1_new <- dwmw(m1, scale = TRUE, max_message_iter = 3)
summary(m1_new)
```

| fstruction | *Formula removal of singular random effects* |

Description

Removes those random effects from a model formula making the model to be singular.

Usage

```
fstruction(model, tol = 1e-4)
```

Arguments

model
: The model of interest, output of either [lmer](#) or [glmer](#), that is, an object of class [merMod](#) and *subclass* either lmerMod or glmerMod.

tol
: A numeric value (default is 1e-4), tolerance under which to determine if the random effects make the model to be singular.

Value

A list with a string component dstring which is the formula as character updated (to be used by update) after removing the singular random effects and, when no all the random effects are removed two other string components:

- betchar: a character vector with the random effect terms removed from the formula.
- inchar: a character vector having the same length as betchar with the corresponding groups (aggregate levels) for which the random effects are removed.

Author(s)

Iago Giné-Vázquez, <iago.gin-vaz@protonmail.com>

See Also

[formula](#), [update](#)

Examples

```
if(requireNamespace("nlme")){
library(lme4)
data(Orthodont, package = "nlme")
Orthodont$nsex <- as.numeric(Orthodont$Sex == "Male")
Orthodont$nsexage <- with(Orthodont, nsex*age)
## The next model is singular
fmo <- lmer(distance ~ age + (age|Subject) + (0+nsex|Subject) +
    (0 + nsexage|Subject), data = Orthodont)
summary(fmo)

## Let's see the formula updated (as a string)
fstruction(fmo)
}
```

---

issue618 *Data example github issue 618 on the lme4 repository*

Description

Three variables, being one dichotomous, other continuous and the last, categorical. Data for the first reproducible example for issue 618 on the lme4 Github-repository, making glmer to produce a

```
Model is nearly unidentifiable: very large eigenvalue -
Rescale variables?
```

warning for the model outcome_dead ~ AGE + (1|ZIP).

Usage

```
data("issue618")
```

Format

A data frame with 1000 observations on the following 3 variables.

outcome_dead an integer vector

AGE an integer vector

ZIP an integer vector

Source

convergence issues with continuous variables in model

Examples

```
data(issue618)
str(issue618)

library(lme4)
m1<- glmer(outcome_dead ~ AGE + (1|ZIP), family = binomial, data = issue618)
summary(m1)
m2 <- dwmw(m1, scale = TRUE)
summary(m2)
```

---

issue618large	*Data example github issue 618 on the lme4 repository*

Description

Three variables, being one dichotomous, other continuous and the last, categorical. Data for the first reproducible example for issue 618 on the lme4 Github-repository, making glmer to produce a

```
Model is nearly unidentifiable: very large eigenvalue -
Rescale variables?
```

warning for the model outcome_dead ~ AGE + (1|ZIP).

Usage

```
data("issue618")
```

Format

A data frame with 1000 observations on the following 3 variables.

outcome_dead an integer vector

AGE an integer vector

ZIP an integer vector

Source

convergence issues with continuous variables in model

Examples

```
data(issue618large)
str(issue618large)

library(lme4)
m1<- glmer(outcome_dead ~ AGE + (1|ZIP), family = binomial, data = issue618large)
summary(m1)
m2 <- dwmw(m1, scale = TRUE)
summary(m2)
m3 <- glmer(outcome_dead ~ scale(AGE) + (1|ZIP), family = binomial, data = issue618large,
    nAGQ=20)
aa <- allFit(m3)
ss <- summary(aa)
ss$msgs  ## all NULL
```

---

plants	*Data for the SO question* R*hrefhttps://stackoverflow.com/questions/60028673/lme4-error-boundary-singular-fit-see-issingularlme4 error: boundary (singular) fit: see ?isSingular.*

Description

Five variables, being one continuous to use as outcome (`Weight`), and four factors, of which two (`Rep` and `PLANT`) are used as clusters for the random effects of a singular linear mixed model (`Weight ~ 1 + (1|Rep:PLANT)`).

Usage

```
data("plants")
```

Format

A data frame with 536 observations on the following 5 variables.

Line a factor with levels Line1 Line10 Line11 Line12 Line13 Line14 Line15 Line16 Line17
     Line18 Line19 Line2 Line20 Line21 Line22 Line23 Line24 Line25 Line26 Line27 Line28
     Line29 Line3 Line30 Line31 Line32 Line33 Line34 Line35 Line36 Line37 Line38 Line39
     Line4 Line40 Line41 Line42 Line43 Line44 Line45 Line46 Line47 Line48 Line49 Line5
     Line50 Line51 Line52 Line53 Line54 Line55 Line56 Line57 Line58 Line59 Line6 Line60
     Line61 Line62 Line63 Line64 Line65 Line66 Line67 Line7 Line8 Line9

Rep a factor with levels 1 2

Weight a numeric vector

PLANT a factor with levels 1 2 3 4

X a factor with levels 24 12 21 11 13 14 22 23

Source

Examples

```
data(plants)
str(plants)

library(lme4)
fit <- lmer(Weight ~ 1 + (1|PLANT:Rep), data = plants)
summary(fit)
fit2 <- dwmw(fit)
summary(fit2)
```

---

treatments	*Data for the Cross Validated question* R*hrefhttps://stats.stackexchange.com/questions/575666/lme4-glmer-warning-messages-with-count-data-mixed-effects-model-and-how-to-prlme4: glmer() warning messages with count data mixed-effects model and how to proceed with model fit*

Description

A continuous variable to be used as outcome (`total_no`), another to be used as predictor (`week`), two factor variables to be used as predictors (`treatment` and `fzone`) and another factor to be used as cluster for the random effects (`plot`) of a Poisson model failing to converge, and an extra variable.

Usage

```
data("treatments")
```

Format

A data frame with 142 observations on the following 7 variables.

plot a numeric vector

date a character vector

total_no a numeric vector

zone a character vector

treatment a character vector

week a numeric vector

fzone a character vector

Source

lme4: glmer() warning messages with count data mixed-effects model and how to proceed with model fit

Examples

```
data(treatments)
str(treatments)

library(lme4)
glmm.1 <- glmer(total_no ~ week*treatment*fzone + (1|plot), data = treatments,
               family = poisson)
summary(glmm.1)
glmm.11 <- dwmw(glmm.1, verbose = TRUE)
summary(glmm.11)
```

Index