

Package ‘glmertree’

June 3, 2017

Title Generalized Linear Mixed Model Trees

Date 2017-05-30

Version 0.1-1

Description Recursive partitioning based on (generalized) linear mixed models (GLMMs) combining lmer()/glmer() from lme4 and lmtree()/glmertree() from partykit.

Depends R (>= 3.1.0), lme4, partykit (>= 1.0-4)

Imports graphics, stats, utils, Formula

Suggests vcd, lattice

LazyData yes

License GPL-2 | GPL-3

NeedsCompilation no

Author Marjolein Fokkema [aut, cre],
Achim Zeileis [aut]

Maintainer Marjolein Fokkema <M.Fokkema@fsw.leidenuniv.nl>

Repository CRAN

Date/Publication 2017-06-03 17:45:25 UTC

R topics documented:

DepressionDemo	2
glmertree	3
Index	6

DepressionDemo

Artificial depression treatment dataset

Description

Simulated dataset of a randomized clinical trial (N = 150) to illustrate fitting of (G)LMM trees.

Usage

```
data("DepressionDemo")
```

Format

A data frame containing 150 observations on 6 variables:

depression numeric. Continuous treatment outcome variable (range: 3-16, M = 9.12, SD = 2.66).

treatment factor. Binary treatment variable.

cluster factor. Indicator for cluster with 10 levels.

age numeric. Continuous partitioning variable (range: 18-69, M = 45, SD = 9.56).

anxiety numeric. Continuous partitioning variable (range: 3-18, M = 10.26, SD = 3.05).

duration numeric. Continuous partitioning variable (range: 1-17, M = 6.97, SD = 2.90).

depression_bin factor. Binarized treatment outcome variable (0 = recovered, 1 = not recovered).

Details

The data were generated such that the duration and anxiety covariates characterized three subgroups with differences in treatment effects. The cluster variable was used to introduce a random intercept that should be accounted for. The treatment outcome is an index of depressive symptomatology.

See Also

[lmertree](#)

Examples

```
data("DepressionDemo", package = "glmertree")
summary(DepressionDemo)
lt <- lmertree(depression ~ treatment | cluster | anxiety + duration + age,
              data = DepressionDemo)
plot(lt)
gt <- glmertree(depression_bin ~ treatment | cluster | anxiety + duration + age,
               data = DepressionDemo)
plot(gt)
```

glmerTree

*(Generalized) Linear Mixed Model Trees***Description**

Model-based recursive partitioning based on (generalized) linear mixed models.

Usage

```
glmerTree(formula, data, weights = NULL,
           ranefstart = NULL, abstol = 0.001, maxit = 100,
           joint = TRUE, dfsplit = TRUE, verbose = FALSE, plot = FALSE,
           lmer.control = lmerControl(), ...)
```

```
glmerTree(formula, data, family = "binomial", weights = NULL,
           ranefstart = NULL, abstol = 0.001, maxit = 100,
           joint = TRUE, dfsplit = TRUE, verbose = FALSE, plot = FALSE,
           glmer.control = glmerControl(), ...)
```

Arguments

formula	formula specifying the response variable and a three-part right-hand-side describing the regressors, random effects, and partitioning variables, respectively. For details see below.
data	data.frame to be used for estimating the model tree.
family	family specification for <code>glmTree</code> and <code>glmer</code> . See glm documentation for families.
weights	numeric. An optional numeric vector of weights. (Note that this is passed with standard evaluation, i.e., it is not enough to pass the name of a column in data.)
ranefstart	numeric. A vector of length <code>nrow(data)</code> , to be used as an offset in estimation of the first tree. NULL by default, which results in a zero offset initialization.
abstol	numeric. The convergence criterion used for estimation of the model. When the difference in log-likelihoods of the random-effects model from two consecutive iterations is smaller than <code>abstol</code> , estimation of the model tree has converged.
maxit	numeric. The maximum number of iterations to be performed in estimation of the model tree.
joint	logical. Should the fixed effects from the tree be (re-)estimated jointly along with the random effects?
dfsplit	logical or numeric. <code>as.integer(dfsplit)</code> is the degrees of freedom per selected split employed when extracting the log-likelihood.
verbose	Should the log-likelihood value of the estimated random-effects model be printed for every iteration of the estimation?
plot	Should the tree be plotted at every iteration of the estimation? Note that selecting this option slows down execution of the function.

`lmer.control`, `glmer.control`
 list. An optional list with control parameters to be passed to `lmer()` and `glmer()`, respectively. See [lmerControl](#) and [glmerControl](#) for details.

...
 Additional arguments to be passed to `lmtree()` or `glmtree()`. See [mob_control](#) documentation for details.

Details

(G)LMM trees learn a tree where each terminal node is associated with different regression coefficients while adjusting for global random effects (such as a random intercept). This allows for detection of subgroup-specific fixed effects, keeping the random effects constant throughout the tree. The estimation algorithm iterates between (1) estimation of the tree given an offset of random effects, and (2) estimation of a random-effects model given the tree structure. See Fokkema et al. (2015) for a detailed introduction.

To specify all variables in the model a formula such as `y ~ x1 + x2 | random | z1 + z2 + z3` is used, where `y` is the response, `x1` and `x2` are the regressors in every node of the tree, `random` is the random effect, and `z1` to `z3` are the partitioning variables considered for growing the tree. If `random` is only a single variable such as `id` a random intercept with respect to `id` is used. Alternatively, it may be an explicit random-effects formula such as `(1 | id)` or a more complicated formula. (Note that in the latter case, the brackets are necessary to protect the pipes in the random effects formulation.)

In the random-effects model from step (2), two strategies are available: Either the fitted values from the tree can be supplied as an offset (`joint = FALSE`) so that only the random effects are estimated. Or the fixed effects are (re-)estimated along with the random effects using a nesting factor with nodes from the tree (`joint = TRUE`). In the former case, the estimation of each random-effects model is typically faster but more iterations are required.

The code is still under development and might change in future versions.

Value

The function returns a list with the following objects:

<code>tree</code>	The final <code>lmtree</code> / <code>glmtree</code> .
<code>lmer</code>	The final <code>lmer</code> random-effects model.
<code>ranef</code>	The corresponding random effects of <code>lmer</code> .
<code>varcorr</code>	The corresponding <code>VarCorr(lmer)</code> .
<code>variance</code>	The corresponding <code>attr(VarCorr(lmer), "sc")^2</code> .
<code>data</code>	The dataset specified with the <code>data</code> argument including added auxiliary variables <code>.ranef</code> and <code>.tree</code> from the last iteration.
<code>loglik</code>	The log-likelihood value of the last iteration.
<code>iterations</code>	The number of iterations used to estimate the <code>lmtree</code> .
<code>maxit</code>	The maximum number of iterations specified with the <code>maxit</code> argument.
<code>ranefstart</code>	The random effects used as an offset, as specified with the <code>ranefstart</code> argument.
<code>formula</code>	The formula as specified with the <code>formula</code> argument.

randomformula	The formula as specified with the randomformula argument.
abstol	The prespecified value for the change in log-likelihood to evaluate convergence, as specified with the abstol argument.
mob.control	A list containing control parameters passed to lmtree(), as specified with
lmer.control	A list containing control parameters passed to lmer(), as specified in the lmer.control argument.
joint	Whether the fixed effects from the tree were (re-)estimated jointly along with the random effects, specified with the joint argument.

References

Fokkema M, Smits N, Zeileis A, Hothorn T, Kelderman H (2015). “Detecting Treatment-Subgroup Interactions in Clustered Data with Generalized Linear Mixed-Effects Model Trees”. Working Paper 2015-10. Working Papers in Economics and Statistics, Research Platform Empirical and Experimental Economics, Universität Innsbruck. <http://EconPapers.RePEc.org/RePEc:inn:wpaper:2015-10>

See Also

[lmer](#), [glmer](#), [lmtree](#), [glmmtree](#)

Examples

```
## artificial example data
data("DepressionDemo", package = "glmtree")

## fit normal linear regression LMM tree for continuous outcome
lt <- lmtree(depression ~ treatment | cluster | age + anxiety + duration,
  data = DepressionDemo)
print(lt)
plot(lt, which = "all") # default behavior, which may also be "tree" or "ranef"
coef(lt)
ranef(lt)
predict(lt, type = "response") # default behavior, type may also be "node"
residuals(lt)

## fit logistic regression GLMM tree for binary outcome
gt <- glmmtree(depression_bin ~ treatment | cluster | age + anxiety + duration,
  data = DepressionDemo)
print(gt)
plot(gt, which = "all") # default behavior, which may also be "tree" or "ranef"
coef(gt)
ranef(gt)
predict(gt, type = "response") # default behavior, type may also be "node" or "link"
residuals(gt)
```

Index

*Topic **datasets**

DepressionDemo, 2

*Topic **tree**

glmertree, 3

as.party.glmertree (glmertree), 3

as.party.lmertree (glmertree), 3

coef.glmertree (glmertree), 3

coef.lmertree (glmertree), 3

DepressionDemo, 2

glm, 3

glmer, 5

glmerControl, 4

glmertree, 3

glmtree, 5

lmer, 5

lmerControl, 4

lmertree, 2

lmertree (glmertree), 3

lmtree, 5

logLik.glmertree (glmertree), 3

logLik.lmertree (glmertree), 3

mob_control, 4

model.frame.glmertree (glmertree), 3

model.frame.lmertree (glmertree), 3

plot.glmertree (glmertree), 3

plot.lmertree (glmertree), 3

predict.glmertree (glmertree), 3

predict.lmertree (glmertree), 3

print.glmertree (glmertree), 3

print.lmertree (glmertree), 3

ranef.glmertree (glmertree), 3

ranef.lmertree (glmertree), 3

resid.glmertree (glmertree), 3

resid.lmertree (glmertree), 3

residuals.glmertree (glmertree), 3

residuals.lmertree (glmertree), 3

terms.glmertree (glmertree), 3

terms.lmertree (glmertree), 3