Package ‘HLMDiag’

February 19, 2015

Type Package
Title Diagnostic tools for hierarchical (multilevel) linear models
Version 0.2.5
Date 2014-1-30
Author Adam Loy <loyad01@gmail.com>
Maintainer Adam Loy <loyad01@gmail.com>
Description A suite of diagnostic tools for hierarchical (multilevel) linear models. The package offers not only leverage and traditional deletion diagnostics (Cook's distance, covratio, covtrace, and MDFFITS) but also provides convenience functions and graphics for residual analysis. The package assumes that models were fit using the lme4 package.
Depends R (>= 2.15.0), lme4 (>= 1.0)
Imports ggplot2 (>= 0.9.2), stats, stats4, methods, plyr, reshape2, MASS, Matrix, Rcpp
LinkingTo Rcpp, RcppArmadillo
Suggests mlmRev, WWGbook
License GPL-2
LazyLoad yes
LazyData yes
Collate 'diagnostic_functions.R' 'group_level_residual_functions.R'
  'identification.R' 'plot_functions.R' 'quantile_functions.R'
  'adjust_formula_lmlist.R' 'case_delete.R' 'LSresids.R'
  'HLMresid.R' 'help.R' 'influence_functions.R'
  'utility_functions.R' 'rotate_ranefs.R' 'autism.R' 'ahd.R'
  'radon.R'
NeedsCompilation yes
Repository CRAN
Date/Publication 2014-01-31 00:47:28
adjust_lmList

Fitting Common Models via lm

Description

Separate linear models are fit via lm similar to lmList, however, adjust_lmList can handle models where a factor takes only one level within a group. In this case, the formula is updated eliminating the offending factors from the formula for that group as the effect is absorbed into the intercept.

Usage

adjust_lmList(formula, data, pool)

Arguments

formula a linear formula such as that used by lmList, e.g. y ~ x1 + ... + xn | g, where g is a grouping factor.
data a data frame containing the variables in the model.
pool a logical value that indicates whether the pooled standard deviation/error should be used.
References

Douglas Bates, Martin Maechler and Ben Bolker (2012). lme4: Linear mixed-effects models using S4 classes. R package version 0.999999-0.

See Also

lmList, lm

Examples

data(Exam, package = 'mlmRev')
seplM <- adjust_lmList(normexam ~ standLRT + sex + schgend | school, data = Exam)
confint(seplM)

---

adjust_lmList-class  Class "adjust_lmList"

Description

Adjusted List of lm Objects with a Common Model

Objects from the Class

Objects can be created by calls of the form new("adjust_lmList", ...) or by adjust_lmList.

Slots

.Data: Object of class "list", a list of lm objects
call: Object of class "call", the call used to create the adjust_lmList object.
pool: Object of class "logical", a logical expression stating whether the pooled standard deviation should be estimate should be used.

Extends

Class "list", from data part. Class "vector", by class "list", distance 2.

Methods

coeff signature(object = "adjust_lmList"): extracts the model coefficients from the individually fit models.
confint signature(object = "adjust_lmList"): computes confidence intervals for the for the parameters of the individually fit models.
show signature(object = "adjust_lmList"): extracts the formula used to create the adjust_lmList object.
plot signature(object = "adjust_lmList.confint"): plot the confidence intervals for the parameters of the individually fit models using the ggplot2 framework.
Methylprednisolone data

Description

Data from a longitudinal study examining the effectiveness of Methylprednisolone as a treatment for patients with severe alcoholic hepatitis. Subjects were randomly assigned to a treatment (31 received a placebo, 35 received the treatment) and serum bilirubin was measured each week for four weeks.

Format

A data frame with 330 observations on the following 5 variables:

- treatment: The treatment a subject received - a factor. Levels are placebo and treated.
- subject: Subject ID - a factor.
- week: Week of the study (0–4) - the time variable.
- sbvalue: Serum bilirubin level (in μmol/L).
- baseline: A subject’s serum bilirubin level at week 0.

Source


References


Autism data

Description

Data from a prospective longitudinal study following 214 children between the ages of 2 and 13 who were diagnosed with either autism spectrum disorder or non-spectrum developmental delays at age 2.
Format

A data frame with 604 observation on the following 7 variables:

- **childid**: Child ID.
- **sicdegp**: Sequenced Inventory of Communication Development group (an assessment of expressive language development) - a factor. Levels are low, med, and high.
- **age2**: Age (in years) centered around age 2 (age at diagnosis).
- **vsae**: Vineland Socialization Age Equivalent
- **gender**: Child’s gender - a factor. Levels are male and female.
- **race**: Child’s race - a factor. Levels are white and nonwhite.
- **bestest2**: Diagnosis at age 2 - a factor. Levels are autism and pdd (pervasive developmental disorder).

Source

[http://www-personal.umich.edu/~kwelch/](http://www-personal.umich.edu/~kwelch/)

References


---

**Case Deletion for mer/lmerMod objects**

Description

This function is used to iteratively delete groups corresponding to the levels of a hierarchical linear model. It uses `lmer()` to fit the models for each deleted case (i.e. uses brute force). To investigate numerous levels of the model, the function will need to be called multiple times, specifying the group (level) of interest each time.

Usage

```r
## Default S3 method:
case_delete(model, ...)

## S3 method for class 'mer'
case_delete(model, group = NULL,
            type = c("both", "fixef", "varcomp"), delete = NULL,
            ...)
```
## case_delete.default

```r
## S3 method for class 'lmerMod'
case_delete(model, group = NULL,
    type = c("both", "fixef", "varcomp"), delete = NULL,
    ...)
```

### Arguments

- **model**: the original hierarchical model fit using `lmer()`
- **group**: a variable used to define the group for which cases will be deleted. If this is left `NULL` (default), then the function will delete individual observations.
- **type**: the part of the model for which you are obtaining deletion diagnostics: the fixed effects ("fixef"), variance components ("varcomp"), or "both" (default).
- **delete**: index of individual cases to be deleted. For higher level units specified in this manner, the `group` parameter must also be specified. If `delete = NULL` then all cases are iteratively deleted.
- **...**: do not use

### Value

A list with the following components:

- **fixef.original**: the original fixed effects estimates
- **ranef.original**: the original predicted random effects
- **vcov.original**: the original variance-covariance matrix for the fixed effects
- **varcomp.original**: the original estimated variance components
- **fixef.delete**: a list of the fixed effects estimated after case deletion
- **ranef.delete**: a list of the random effects predicted after case deletion
- **vcov.delete**: a list of the variance-covariance matrices for the fixed effects obtained after case deletion
- **fitted.delete**: a list of the fitted values obtained after case deletion
- **varcomp.delete**: a list of the estimated variance components obtained after case deletion

### Author(s)

Adam Loy <loyad01@gmail.com>

### References


**Examples**

```r
data(sleepstudy, package = 'lme4')
fm <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy)

# Deleting every Subject
fmDel <- case_delete(model = fm, group = "Subject", type = "both")

# Deleting only subject 308
del308 <- case_delete(model = fm, group = "Subject", type = "both", delete = 308)

# Deleting a subset of subjects
delSubset <- case_delete(model = fm, group = "Subject", type = "both", delete = 308:310)
```

---

**Description**

This function creates a plot (using `qplot()`) where the shrinkage estimate appears on the horizontal axis and the LS estimate appears on the vertical axis.

**Usage**

```r
compare_eb_ls(eb, ols, identify = FALSE, silent = TRUE, 
              ...)```

**Arguments**

- `eb`: a matrix of random effects
- `ols`: a matrix of the OLS estimates found using `random_ls_coef`
- `identify`: the percentage of points to identify as unusual, `FALSE` if you do not want the points identified.
- `silent`: logical: should the list of data frames used to make the plots be suppressed.
- `...`: other arguments to be passed to `qplot()`

**Author(s)**

Adam Loy <loyad01@gmail.com>

**Examples**

```r
wages.fm1 <- lmer(lnw ~ exper + (exper | id), data = wages)
wages.sepLM <- adjust lmList(lnw ~ exper | id, data = wages)
rancoef.eb <- coef(wages.fm1)$id
rancoef.ols <- coef(wages.sepLM)
compare_eb_ls(eb = rancoef.eb, ols = rancoef.ols, identify = 0.01)
```
covratio.default

Description

These functions calculate measures of the change in the covariance matrices for the fixed effects based on the deletion of an observation, or group of observations, for a hierarchical linear model fit using `lmer`.

Usage

```r
## Default S3 method:
covratio(object, ...)

## Default S3 method:
covtrace(object, ...)

## S3 method for class 'mer'
covratio(object, group = NULL,
          delete = NULL, ...)

## S3 method for class 'lmerMod'
covratio(object, group = NULL,
          delete = NULL, ...)

## S3 method for class 'mer'
covtrace(object, group = NULL,
          delete = NULL, ...)

## S3 method for class 'lmerMod'
covtrace(object, group = NULL,
          delete = NULL, ...)
```

Arguments

- `object`: fitted object of class `mer` or `lmerMod`
- `group`: variable used to define the group for which cases will be deleted. If `group = NULL`, then individual cases will be deleted.
- `delete`: index of individual cases to be deleted. To delete specific observations the row number must be specified. To delete higher level units the group ID and `group` parameter must be specified. If `delete = NULL` then all cases are iteratively deleted.
- `...`: do not use
Details

Both the covariance ratio (covratio) and the covariance trace (covtrace) measure the change in the covariance matrix of the fixed effects based on the deletion of a subset of observations. The key difference is how the variance covariance matrices are compared: covratio compares the ratio of the determinants while covtrace compares the trace of the ratio.

Value

If delete = NULL then a vector corresponding to each deleted observation/group is returned. If delete is specified then a single value is returned corresponding to the deleted subset specified.

Author(s)

Adam Loy <loyad01@gmail.com>

References


See Also

leverage.mer, cooks.distance.mer, mdffits.mer, rvc.mer

Examples

```r
# data(sleepstudy, package = 'lme4')
ss <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)

# covratio for individual observations
ss.cr1 <- covratio(ss)

# covratio for subject-level deletion
ss.cr2 <- covratio(ss, group = "Subject")

## Not run:
## A larger example
data(Exam, package = 'mlmRev')
fm <- lmer(normexam ~ standLRT * schavg + (standLRT | school), Exam)

# covratio for individual observations
cr1 <- covratio(fm)

# covratio for school-level deletion
cr2 <- covratio(fm, group = "school")

## End(Not run)
# covtrace for individual observations
```
Calculating influence diagnostics for HLMs.

Description

This group of functions is used to compute deletion diagnostics for a hierarchical linear model based on the building blocks returned by `case_delete`.

Usage

diagnostics(object)

## S3 method for class 'case_delete'
cooks.distance(model, ...)

## S3 method for class 'case_delete'
mdffits(object, ...)

## S3 method for class 'case_delete'
covtrace(object, ...)

## S3 method for class 'case_delete'
covratio(object, ...)

## S3 method for class 'case_delete'
rvc(object, ...)

Arguments

object an object containing the output returned by `case_delete`
model an object containing the output returned by `case_delete`. This is only named differently to agree with the generic.
... do not use
diagnostics

Details

The primary function is diagnostics which returns either a list or data frame of influence measures depending on whether type = "both" (list) or if only one aspect of the model is selected (data.frame). If type = "both", then a list with Cook’s distance, MDFFITS, COVTRACE, and COVRATIO are returned for the fixed effects and relative variance change (RVC) is returned for the variance components.

The methods cooks.distance, mdffits, covtrace, covratio, and rvc can be used for direct computation of the corresponding diagnostic quantities from an object of class case_delete.

Note

The results provided by this function will give exact values of the diagnostics; however, these are computationally very slow. Approximate versions of cooks.distance, mdffits, covtrace, covratio are implemented in HLMdiag, and can be called directly on the mer object.

Author(s)

Adam Loy <loyad01@gmail.com>

References


See Also

leverage.mer, cooks.distance.mer, mdffits.mer, covratio.mer, covtrace.mer

Examples

data(sleepstudy, package = 'lme4')
fm <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)

# Subject level deletion and diagnostics
subject.del <- case_delete(model = fm, group = "Subject", type = "both")
subject.diag <- diagnostics(subject.del)
Description

This is a function that can be used to create (modified) dotplots for the diagnostic measures. The plot allows the user to understand the distribution of the diagnostic measure and visually identify unusual cases.

Usage

dotplot_diag(x, index, data, cutoff, name = c("cooks.distance", "mdffits", "covratio", "covtrace", "rvc", "leverage"), modify = FALSE, ...)

Arguments

x values of the diagnostic of interest
index index (IDs) of x values
data data frame to use (optional)
cutoff value(s) specifying the boundary for unusual values of the diagnostic. The cutoff(s) can either be supplied by the user, or automatically calculated using measures of internal scaling if cutoff = "internal"
name what diagnostic is being plotted (one of "cooks.distance", "mdffits", "covratio", "covtrace", "rvc", or "leverage"). this is used for the calculation of "internal" cutoffs
modify specifies the geom to be used to produce a space-saving modification: either "dotplot" or "boxplot"
... other arguments to be passed to qplot()

Note

The resulting plot uses coord_flip to rotate the plot, so when adding customized axis labels you will need to flip the usage of xlab and ylab.

Author(s)

Adam Loy <loyad01@gmail.com>

Examples

data(sleepstudy, package = 'lme4')
fm <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)

# Subject level deletion and diagnostics
subject.del <- case_delete(model = fm, group = "Subject", type = "both")
ggplot_qqnorm

Constructing a normal Q-Q plot

Description

This function will construct a normal Q-Q plot within the ggplot2 framework. It combines the functionality of qqnorm and qqline.

Usage

ggplot_qqnorm(x, line = NULL, ...)

Arguments

x a numeric vector

line the method used to fit a reference line. If no reference line is desired, leave the value as NULL. line = "rlm" will use robust regression to fit a reference line. line = "quantile" will fit a line through the first and third quartiles. These options are the same as those given for the qplot function in the car package.

... other arguments to be passed to qplot()

Author(s)

Adam Loy <loyad01@gmail.com>

See Also

qqnorm, qqline
Description

This function will overlay multiple normal Q-Q plots on the same plot. This will be particularly useful when comparing the distribution between groups. In this situation, significantly different slopes would indicate the normal distributions for the groups do not share a common standard deviation.

Usage

group_qqnorm(x, group, line = NULL, alpha_point = 1,
alpha_line = 1, ...)

Arguments

x a numeric vector from which quantiles will be calculated

group a vector indicating group membership for each value in x.

line the method used to fit reference lines. If no reference lines are desired, leave the value as NULL. line = "rlm" will use robust regression to fit reference lines. line = "quantile" will fit lines through the first and third quartiles.

... other arguments to be passed to ggplot

alpha_point alpha value specified for the points

alpha_line alpha value specified for the lines

Author(s)

Adam Loy <loyad01@gmail.com>

References

Description

HLMdiag provides a suite of diagnostic tools for hierarchical (multilevel) linear models fit using \texttt{lmer}. These tools are grouped below by purpose. See the help documentation for additional information about each function.

Details

Residual analysis

HLMdiag’s \texttt{HLMresid} function provides a convenient wrapper to obtain residuals at each level of a hierarchical linear model. In addition to being a wrapper function for functions implemented in the \texttt{lme4} package, HLMresid provides access to the marginal and least squares residuals (through \texttt{LSresids}) that were not previously implemented.

Influence analysis

Influence on fitted values

HLMdiag provides \texttt{leverage} that calculates the influence that observations/groups have on the fitted values (leverage). For mixed/hierarchical models leverage can be decomposed into two parts: the fixed part and the random part. We refer the user to the references cited in the help documentation for additional explanation.

Influence on fixed effects estimates

HLMdiag provides \texttt{cooks.distance} and \texttt{mdffits} to assess the influence of subsets of observations on the fixed effects.

Influence on precision of fixed effects

HLMdiag provides \texttt{covratio} and \texttt{covtrace} to assess the influence of subsets of observations on the precision of the fixed effects.

Influence on variance components

HLMdiag’s \texttt{rvc} calculates the relative variance change to assess the influence of subsets of observations on the variance components.

Graphics

HLMdiag also strives to make graphical assessment easier in the \texttt{ggplot2} framework by providing dotplots for influence diagnostics (\texttt{dotplot.diag}), grouped Q-Q plots (\texttt{group_qqnorm}), and Q-Q plots that combine the functionality of \texttt{qqnorm} and \texttt{qline} (\texttt{ggplot_qqnorm}).
HLMresid.default Calculating residuals from HLMs

Description

HLMresid is a function that extracts residuals from a hierarchical linear model fit using lmer. That is, it is a unified framework that extracts/calculates residuals from mer or lmerMod objects.

Usage

```r
## Default S3 method:
HLMresid(object, ...)

## S3 method for class 'mer'
HLMresid(object, level, type = "EB",
         sim = NULL, standardize = FALSE, ...)

## S3 method for class 'lmerMod'
HLMresid(object, level, type = "EB",
         sim = NULL, standardize = FALSE, ...)
```

Arguments

- `object`: an object of class mer or lmerMod.
- `level`: which residuals should be extracted: 1 for within-group (case-level) residuals, the name of a grouping factor (as defined in flist of the mer object) for between-group residuals, or marginal.
- `type`: how are the residuals predicted: either "EB" or "LS" (the default is "EB").
- `sim`: optional argument giving the data frame used for LS residuals. This is used mainly for dealing with simulations.
- `standardize`: if standardize = TRUE the standardized residuals will be returned; if standardize = "semi" then the semi-standardized level-1 residuals will be returned. Note that for higher-level residuals of type = "LS", standardize = TRUE does not result in standardized residuals as they have not been implemented.
- `...`: do not use

Details

This function extracts residuals from the model, and can extract residuals estimated using least squares (LS) or Empirical Bayes (EB). This unified framework enables the analyst to more easily conduct an upward residual analysis during model exploration/checking.

The HLMresid function provides a wrapper that will extract residuals from a fitted mer or lmerMod object. The function provides access to residual quantities already made available by the functions resid and ranef, but adds additional functionality. Below is a list of types of residuals that can be extracted.
raw level-1 residuals These are equivalent to the residuals extracted by resid if level = 1, type = "EB", and standardize = FALSE is specified. You can also specify type = "LS" for LS residuals that are not equivalent to those from resid.

standardized level-1 residuals Specify level = 1, and standardize = TRUE. This works with both type = "EB" or "LS".

semi-standardized level-1 residuals Specify level = 1, type = "LS" and standardize = "semi".

raw group level residuals These are equivalent to extracting the predicted random effects for a given group using ranef. Set level to a grouping factor name and type = "EB". type = "LS" can also be specified, though this is less common.

standardized group level residuals Set level to a grouping factor name, type = "EB", and standardized = TRUE. This will not produce standardized residuals for type = "LS".

marginal residuals The marginal residuals can be obtained by setting level = "marginal". Only type = "EB" is implemented.

cholesky residuals These are essentially standardized marginal residuals. To obtain cholesky residuals set level = "marginal", type = "EB", and standardize = TRUE.

Note that standardize = "semi" is only implemented for level-1 LS residuals.

Author(s)
Adam Loy <loyad01@gmail.com>

References


See Also
LSresids, resid, ranef

Examples
data(sleepstudy, package = "lme4")
fml <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy)

# level-1 residuals
all.equal(HLMresid(object = fml, level = 1, type = "EB"), resid(fml)) ## EB
rLS <- HLMresid(object = fml, level = 1, type = "LS") ## raw LS resids
head(rLS)
rLS.std <- HLMresid(object = fml, level = 1, type = "LS", standardize = TRUE) ## std. LS resids
head(rLS.std)

# level-2 residuals
all.equal(r2EB <- HLMresid(object = fml, level = "Subject", type = "EB"), ranef(fml)[["Subject"]])
r2EB.std <- HLMresid(object = fml, level = "Subject", type = "EB", standardize = TRUE)
leverage.default

head(r2EB)
head(r2EB.std)

# marginal residuals
mr <- HLMresid(object = fml, level = "marginal")
cholr <- HLMresid(object = fml, level = "marginal", standardize = TRUE) # Cholesky residuals

leverage.default  Leverage for HLMs

Description
This function calculates the leverage of a hierarchical linear model fit by lmer.

Usage

## Default S3 method:
leverage(object, ...)

## S3 method for class 'mer'
leverage(object, level, ...)

## S3 method for class 'lmerMod'
leverage(object, level, ...)

Arguments

object  fitted object of class mer of lmerMod
level  the level at which the leverage should be calculated: either 1 for observation level leverage or the name of the grouping factor (as defined in flist of the mer object) for group level leverage. leverage assumes that the grouping factors are unique; thus, if IDs are repeated within each unit, unique IDs must be generated by the user prior to use of leverage.
...  do not use

Details

Demidenko and Stukel (2005) describe leverage for mixed (hierarchical) linear models as being the sum of two components, a leverage associated with the fixed ($H_1$) and a leverage associated with the random effects ($H_2$) where

$$H_1 = X(X'V^{-1}X)^{-1}X'V^{-1}$$

and

$$H_2 = ZDZ'V^{-1}(I - H_1)$$

Nobre and Singer (2011) propose using

$$H_2^* = ZDZ'$$
leverage.default

as the random effects leverage as it does not rely on the fixed effects.

For individual observations leverage uses the diagonal elements of the above matrices as the measure of leverage. For higher-level units, leverage uses the mean trace of the above matrices associated with each higher-level unit.

Value

leverage returns a data frame with the following columns:

- overall The overall leverage, i.e. \( H = H_1 + H_2 \).
- fixef The leverage corresponding to the fixed effects.
- ranef The leverage corresponding to the random effects proposed by Demidenko and Stukel (2005).
- ranef.uc The (unconfounded) leverage corresponding to the random effects proposed by Nobre and Singer (2011).

Author(s)

Adam Loy <loyad01@gmail.com>

References


See Also

cooks.distance.mer, mdffits.mer, covratio.mer, covtrace.mer, rvc.mer

Examples

data(sleepstudy, package = 'lme4')
fm <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)

# Observation level leverage
lev1 <- leverage(fm, level = 1)
head(lev1)

# Group level leverage
lev2 <- leverage(fm, level = "Subject")
head(lev2)
Calculating least squares residuals

Description

This function calculates least squares (LS) residuals found by fitting separate LS regression models to each case. For examples see the documentation for HLMresid.

Usage

```r
## Default S3 method:
LSresids(object, ...)

## S3 method for class 'mer'
LSresids(object, level, sim = NULL,
          standardize = FALSE, ...)

## S3 method for class 'lmerMod'
LSresids(object, level, sim = NULL,
          standardize = FALSE, ...)
```

Arguments

- `object`: an object of class `mer` or `lmerMod`.
- `level`: which residuals should be extracted: 1 for case-level residuals or the name of a grouping factor (as defined in `flist` of the `mer` object) for between-group residuals.
- `sim`: optional argument giving the data frame used for LS residuals. This is used mainly when dealing with simulations.
- `standardize`: if TRUE the standardized level-1 residuals will also be returned (if `level = 1`); if "semi" then the semi-standardized level-1 residuals will be returned.
- `...`: do not use

Author(s)

Adam Loy <loyad01@gmail.com>

References


See Also

HLMresid
**Description**

These functions calculate measures of the change in the fixed effects estimates based on the deletion of an observation, or group of observations, for a hierarchical linear model fit using `lmer`.

**Usage**

```r
## Default S3 method:
mdffits(object, ...)

## S3 method for class 'mer'
cooks.distance(model, group = NULL,
               delete = NULL, ...)

## S3 method for class 'lmerMod'
cooks.distance(model, group = NULL,
               delete = NULL, ...)

## S3 method for class 'mer'
mdffits(object, group = NULL,
         delete = NULL, ...)

## S3 method for class 'lmerMod'
mdffits(object, group = NULL,
         delete = NULL, ...)
```

**Arguments**

- `model`: fitted model of class `mer` or `lmerMod`
- `group`: variable used to define the group for which cases will be deleted. If `group = NULL`, then individual cases will be deleted.
- `delete`: index of individual cases to be deleted. To delete specific observations the row number must be specified. To delete higher level units the group ID and group parameter must be specified. If `delete = NULL` then all cases are iteratively deleted.
- `...`: do not use
- `object`: fitted object of class `mer` or `lmerMod`

**Details**

Both Cook's distance and MDFFITS measure the change in the fixed effects estimates based on the deletion of a subset of observations. The key difference between the two diagnostics is that Cook’s distance uses the covariance matrix for the fixed effects from the original model while MDFFITS uses the covariance matrix from the deleted model.
Value

Both functions return a numeric vector (or single value if delete has been specified) with attribute beta_cdd giving the difference between the full and deleted parameter estimates.

Note

Because MDFFITS requires the calculation of the covariance matrix for the fixed effects for every model, it will be slower.

Author(s)

Adam Loy <loyad01@gmail.com>

References


See Also

leverage.mer, covratio.mer, covtrace.mer, rvc.mer

Examples

data(sleepstudy, package = 'lme4')
ss <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)
# Cook's distance for individual observations
ss.cd.lev1 <- cooks.distance(ss)

# Cook's distance for each Subject
ss.cd.subject <- cooks.distance(ss, group = "Subject")

## Not run:
data(Exams, package = 'mlmRev')
fm <- lmer(normexam ~ standlrt * schavg + (standlrt | school), Exams)
# Cook's distance for individual observations
cd.lev1 <- cooks.distance(fm)

# Cook's distance for each school
cd.school <- cooks.distance(fm, group = "school")

# Cook's distance when school 1 is deleted
cd.school1 <- cooks.distance(fm, group = "school", delete = 1)

## End(Not run)
# MDFFITS for individual observations
radon <- read.csv("../data/radon.data")

ss.m1 <- mdffits(ss)

# MDFFITS for each Subject
ss.m.subject <- mdffits(ss, group = "Subject")

## Not run:

# MDFFITS for individual observations
m1 <- mdffits(fm)

# MDFFITS for each school
m.school <- mdffits(fm, group = "school")

## End(Not run)

radon

---

## Description

Radon measurements of 919 owner-occupied homes in 85 counties of Minnesota.

## Format

A data frame with 919 observations on the following 5 variables:

- **log.radon**: Radon measurement (in log pCi/L, i.e., log picoCurie per liter)
- **basement**: Indicator for the level of the home at which the radon measurement was taken - 0 = basement, 1 = first floor.
- **uranium**: Average county-level soil uranium content.
- **county**: County ID.
- **county.name**: County name - a factor.

## Source


## References


**rotate_ranef.default**  
*Calculate s-dimensional rotated random effects*

**Description**

This function calculates reduced dimensional rotated random effects. The rotation reduces the influence of the residuals from other levels of the model so that distributional assessment of the resulting random effects is possible.

**Usage**

```r
## Default S3 method:
rotate_ranef(.mod, ...)

## S3 method for class 'mer'
rotate_ranef(.mod, .L, s = NULL,
    .varimax = FALSE, ...)

## S3 method for class 'lmerMod'
rotate_ranef(.mod, .L, s = NULL,
    .varimax = FALSE, ...)
```

**Arguments**

- `.mod`  
an object of class `mer` or `lmerMod`.
- `.L`  
a matrix defining which combination of random effects are of interest.
- `s`  
the dimension of the subspace of interest.
- `.varimax`  
if `.varimax = TRUE` then the raw varimax rotation will be applied to the resulting rotation.
- `...`  
do not use

**Author(s)**

Adam Loy <loyad01@gmail.com>

---

**rvc.default**  
*Relative variance change for HLMs*

**Description**

This function calculates the relative variance change (RVC) of hierarchical linear models fit via `lmer`.
Usage

```r
## Default S3 method:
rvc(object, ...)

## S3 method for class 'mer'
rvc(object, group = NULL, delete = NULL, ...

## S3 method for class 'lmerMod'
rvc(object, group = NULL, delete = NULL, ...)
```

Arguments

- `object`: fitted object of class `mer` or `lmerMod`
- `group`: variable used to define the group for which cases will be deleted. If `group = NULL`, then individual cases will be deleted.
- `delete`: index of individual cases to be deleted. To delete specific observations the row number must be specified. To delete higher level units the group ID and `group` parameter must be specified. If `delete = NULL` then all cases are iteratively deleted.
- `...`: do not use

Value

- If `delete = NULL` a matrix with columns corresponding to the variance components of the model and rows corresponding to the deleted observation/group is returned.
- If `delete` is specified then a named vector is returned.
- The residual variance is named `sigma2` and the other variance components are named `D**` where the trailing digits give the position in the covariance matrix of the random effects.

Author(s)

Adam Loy <loyad01@gmail.com>

References


See Also

`leverage.mer`, `cooks.distance.mer`, `mdffits.mer`, `covratio.mer`, `covtrace.mer`
varcomp.mer  

Extracting variance components

Description

This function extracts the variance components from a mixed/hierarchical linear model fit using lmer.

Usage

varcomp.mer(object)

Arguments

object  
a fitted model object of class mer or lmerMod.

Value

A named vector is returned. sigma2 denotes the residual variance. The other variance components are names D** where the trailing digits specify the of that variance component in the covariance matrix of the random effects.

Author(s)

Adam Loy <loyad01@gmail.com>

Examples

data(sleepstudy, package = "lme4")
fm1 <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy)
varcomp.mer(fm1)

wages  

Wages for male high school dropouts

Description

Data on the labor-market experience of male high school dropouts.
Wages

Format

A data frame with 6402 observations on the following 15 variables.

- **id**: respondent id - a factor with 888 levels.
- **lnw**: natural log of wages expressed in 1990 dollars.
- **exper**: years of experience in the work force
- **ged**: equals 1 if respondent has obtained a GED as of the time of survey, 0 otherwise
- **postexp**: labor force participation since obtaining a GED (in years) - before a GED is earned postexp = 0, and on the day a GED is earned postexp = 0
- **black**: factor - equals 1 if subject is black, 0 otherwise
- **hispanic**: factor - equals 1 if subject is hispanic, 0 otherwise
- **hgc**: highest grade completed - takes integers 6 through 12
- **hgc.9**: hgc - 9, a centered version of hgc
- **u-rate**: local area unemployment rate for that year
- **ue.7**:
- **ue.center1**:
- **ue.mean**:
- **ue.person.cen**:
- **ue.1**:

Source

These data are originally from the 1979 National Longitudinal Survey on Youth (NLSY79) that can be found here [http://www.bls.gov/nls/nlsdata.htm](http://www.bls.gov/nls/nlsdata.htm).

Singer and Willett (2003) used these data for examples in chapter (insert info. here) and the data sets used can be found on the UCLA Statistical Computing website: [http://www.ats.ucla.edu/stat/examples/alda/](http://www.ats.ucla.edu/stat/examples/alda/)

Additionally the data were discussed by Cook and Swayne (2003) and the data can be found on the GGobi website: [http://www.ggobi.org/book/](http://www.ggobi.org/book/).

References


Examples

```R
str(wages)
summary(wages)
lmer(lnw ~ exper + (exper | id), data = wages)
```
Index

*Topic classes
  adjust.lmList-class, 3
*Topic datasets
  ahd, 4
  autism, 4
  radon, 23
  wages, 26
*Topic hplot
  dotplot_diag, 12
  ggplot_qqnorm, 13
group_qqnorm, 14
*Topic models
  adjust.lmList, 2
  case_delete.default, 5
covratio.default, 8
diagnostics, 10
HLMresid.default, 16
leverage.default, 18
LSresids.default, 20
mdffits.default, 21
rvc.default, 24
varcomp.mer, 26
*Topic package
  HLMdiag, 15
*Topic regression
  adjust.lmList, 2
  case_delete.default, 5
covratio.default, 8
diagnostics, 10
HLMresid.default, 16
leverage.default, 18
LSresids.default, 20
mdffits.default, 21
rvc.default, 24
varcomp.mer, 26

adjust.lmList, 2
adjust.lmList.formula.data.frame-method
  (adjust.lmList), 2
adjust.lmList-class, 3

ahd, 4
autism, 4
case_delete (case_delete.default), 5
case_delete.default, 5
coeff, adjust.lmList-method
  (adjust.lmList-class), 3
compare_eb_ls, 7
confint, adjust.lmList-method
  (adjust.lmList-class), 3
cooks.distance, 15
cooks.distance (mdffits.default), 21
cooks.distance.case_delete
diagnostics, 10
cooks.distance.mer, 9, 11, 19, 25
covratio, 15
covratio (covratio.default), 8
covratio.case_delete (diagnostics), 10
covratio.default, 8
covratio.mer, 11, 19, 22, 25
covtrace, 15
covtrace (covratio.default), 8
covtrace.case_delete (diagnostics), 10
covtrace.mer, 11, 19, 22, 25
diagnostics, 10
dotplot_diag, 12, 15
ggplot_qqnorm, 13, 15
group_qqnorm, 14, 15
HLMdiag, 15
HLMdiag-package (HLMdiag), 15
HLMresid, 15, 20
HLMresid (HLMresid.default), 16
HLMresid.default, 16
leverage, 15
leverage (leverage.default), 18
leverage.default, 18
leverage.mer, 9, 11, 22, 25
list, 3
lm, 3
lmer, 15
lmList, 3
LSresids, 15, 17
LSresids (LSresids.default), 20
LSresids.default, 20

mdffits, 15
mdffits (mdffits.default), 21
mdffits.case_delete (diagnostics), 10
mdffits.default, 21
mdffits.mer, 9, 11, 19, 25

package-HLMdiag (HLMdiag), 15
plot, adjust_lmlist.confint, ANY-method
   (adjust_lmlist-class), 3

qcline, 13, 15
qqnorm, 13, 15

radon, 23
ranef, 17
resid, 17
rotate_ranef (rotate_ranef.default), 24
rotate_ranef.default, 24
rvc, 15
rvc (rvc.default), 24
rvc.case_delete (diagnostics), 10
rvc.default, 24
rvc.mer, 9, 19, 22

show, adjust_lmlist-method
   (adjust_lmlist-class), 3

varcomp.mer, 26
vector, 3

wages, 26