Package: HLMdiag (via r-universe)

October 12, 2026

Type Package

Title Diagnostic Tools for Hierarchical (Multilevel) Linear Models

Version 0.5.1

Maintainer Adam Loy <loyad01@gmail.com>

Description A suite of diagnostic tools for hierarchical (multilevel) linear models. The tools include not only leverage and traditional deletion diagnostics (Cook's distance, covratio, covtrace, and MDFFITS) but also convenience functions and graphics for residual analysis. Models can be fit using either lmer in the 'lme4' package or lme in the 'nlme' package.

Depends R (>= 3.5.0)

Imports ggplot2 (>= 0.9.2), stats, methods, plyr, reshape2, MASS, Matrix, mgcv, dplyr, magrittr, stringr, purrr, tibble, tidyselect, janitor, Rcpp, rlang, ggrepel, diagonals

LinkingTo Rcpp, RcppArmadillo

Suggests mlmRev, WWGbook, lme4 (>= 1.0), nlme, testthat, knitr, rmarkdown, car, gridExtra, qqplotr

License GPL-2

LazyLoad yes

LazyData yes

Collate 'diagnostic_functions.R' 'group_level_residual_functions.R' 'identification.R' 'plot_functions.R' 'adjust_formula_lmList.R' 'case_delete.R' 'LSresids.R' 'HLMdiag-deprecated.R' 'HLMdiag-package.R' 'hlm_augment.R' 'hlm_influence.R' 'hlm_resid.R' 'help.R' 'influence_functions.R' 'utility_functions.R' 'rotate_ranefs.R' 'autism.R' 'ahd.R' 'radon.R' 'pull_resid.R' 'residual_functions.R' 'HLMdiag-defunct.R'

Encoding UTF-8

URL https://github.com/aloy/HLMdiag

BugReports https://github.com/aloy/HLMdiag/issues

RoxygenNote 7.3.2

VignetteBuilder knitr

NeedsCompilation yes

Author Adam Loy [cre, aut], Jaylin Lowe [aut], Jack Moran [aut]

Repository https://cran.r-universe.dev

Date/Publication 2025-05-12 23:20:02 UTC

RemoteUrl https://github.com/cran/HLMdiag

RemoteRef HEAD

RemoteSha 284085a1c42c53a02d40e0c4c36460a0f92c1208

Contents

adjust_lmList.formula . . . 3
ahd . . . 4
autism . . . 4
case_delete.default . . . 5
compare_eb_ls . . . 8
cooks.distance.case_delete . . . 8
covratio.default . . . 11
dotplot_diag . . . 13
extract_design . . . 14
hlm_augment . . . 15
hlm_resid.default . . . 16
HLMdiag . . . 19
HLMdiag-defunct . . . 20
HLMdiag-deprecated . . . 21
leverage.default . . . 21
LSresids.default . . . 23
mdffits.default . . . 24
pull_resid.default . . . 26
radon . . . 28
resid_conditional.default . . . 28
resid_marginal.default . . . 30
resid_ranef . . . 31
rotate_ranef.default . . . 31
rvc.default . . . 32
varcomp.mer . . . 34
wages . . . 34

Index **36**

`adjust_lmList.formula` *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

```
## S3 method for class 'formula'
adjust_lmList(object, data, pool)
```

Arguments

`object`	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)
```

ahd *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 measures each week for four weeks.

Usage

```
data(ahd)
```

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

Vonesh, E. F. and Chinchilli, V. M. (1997) *Linear and Nonlinear Models for the Analysis of Repeated Measurements*. Marcel Dekker, New York.

References

Carithers, R. L., Herlong, H. F., Diehl, A. M., Shaw, E. W., Combes, B., Fallon, H. J. & Maddrey, W. C. (1989) Methylprednisolone therapy in patients with severe alcoholic hepatitis. *Annals of Internal Medicine*, **110**(9), 685–690.

autism *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.

Usage

```
data(autism)
```

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/

References

Anderson, D. K., Lord, C., Risi, S., DiLavore, P. S., Shulman, C., Thurm, A., et al. (2007). Patterns of growth in verbal abilities among children with autism spectrum disorder. *Journal of Consulting and Clinical Psychology*, **75**(4), 594–604.

Anderson, D. K., Oti, R. S., Lord, C., & Welch, K. (2009). Patterns of Growth in Adaptive Social Abilities Among Children with Autism Spectrum Disorders. *Journal of Abnormal Child Psychology*, **37**(7), 1019–1034.

`case_delete.default` *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

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

## S3 method for class 'mer'
case_delete(
  model,
  level = 1,
  type = c("both", "fixef", "varcomp"),
  delete = NULL,
  ...
)

## S3 method for class 'lmerMod'
case_delete(
  model,
  level = 1,
  type = c("both", "fixef", "varcomp"),
  delete = NULL,
  ...
)

## S3 method for class 'lme'
case_delete(
  model,
  level = 1,
  type = c("both", "fixef", "varcomp"),
  delete = NULL,
  ...
)
```

Arguments

`model`	the original hierarchical model fit using `lmer()`
`...`	do not use
`level`	a variable used to define the group for which cases will be deleted. If `level = 1` (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`	numeric index of individual cases to be deleted. If the `level` parameter is specified, `delete` may also take the form of a character vector consisting of group names as they appear in `flist`. It is possible to set `level` and delete individual cases from different groups using `delete`, so numeric indices should be double checked to confirm that they encompass entire groups. If `delete = NULL` then all cases are iteratively deleted.

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

Christensen, R., Pearson, L.M., and Johnson, W. (1992) Case-Deletion Diagnostics for Mixed Models, *Technometrics*, **34**, 38 – 45.

Schabenberger, O. (2004) Mixed Model Influence Diagnostics, in *Proceedings of the Twenty-Ninth SAS Users Group International Conference*, SAS Users Group International.

Examples

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

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

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

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

compare_eb_ls *Visually comparing shrinkage and LS estimates*

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

```
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

```
wages.fm1 <- lme4::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)
```

cooks.distance.case_delete *Calculating influence diagnostics for HLMs*

Description

This function is used to compute influence diagnostics for a hierarchical linear model. It takes a model fit as a `lmerMod` object or as a `lme` object and returns a tibble with Cook's distance, MDFFITS, covtrace, covratio, and leverage.

Usage

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

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

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

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

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

hlm_influence(model, ...)

## Default S3 method:
hlm_influence(model, ...)

## S3 method for class 'lmerMod'
hlm_influence(
  model,
  level = 1,
  delete = NULL,
  approx = TRUE,
  leverage = "overall",
  data = NULL,
  ...
)

## S3 method for class 'lme'
hlm_influence(
  model,
  level = 1,
  delete = NULL,
  approx = TRUE,
  leverage = "overall",
  ...
)
```

Arguments

model	an object of class lmerMod or lme
...	not in use
level	used to define the group for which cases are deleted and influence diagnostics are calculated. If level = 1 (default), then influence diagnostics are calculated

for individual observations. Otherwise, level should be the name of a grouping factor as defined in flist for a lmerMod object or as in groups for a lme object.

delete — numeric index of individual cases to be deleted. If the level parameter is specified, delete may also take the form of a character vector consisting of group names as they appear in flist for lme4 models or as in groups for nlme models. If delete = NULL then all cases are iteratively deleted.

approx — logical parameter used to determine how the influence diagnostics are calculated. If FALSE (default), influence diagnostics are calculated using a one step approximation. If TRUE, influence diagnostics are calculated by iteratively deleting groups and refitting the model using lmer. This method is more accurate, but slower than the one step approximation. If approx = FALSE, the returned tibble also contains columns for relative variance change (RVC).

leverage — a character vector to determine which types of leverage should be included in the returned tibble. There are four options: 'overall' (default), 'fixef', 'ranef', or 'ranef.uc'. One or more types may be specified. For additional information about the types of leverage, see ?leverage.

data — (optional) the data frame used to fit the model. This is only necessary for lmerMod models if na.action = "na.exclude" was set.

Details

The hlm_influence function provides a wrapper that appends influence diagnostics to the original data. The approximated influence diagnostics returned by this function are equivalent to those returned by cooks.distance, mdffits, covtrace, covratio, and leverage. The exact influence diagnostics obtained through a full refit of the data are also available through case_delete and the accompanying functions cooks.distance, mdffits, covtrace, and covratio that can be called directly on the case_delete object.

Note

It is possible to set level and delete individual cases from different groups using delete, so numeric indices should be double checked to confirm that they encompass entire groups. Additionally, if delete is specified, leverage values are not returned in the resulting tibble.

Examples

```
## Not run:
data(sleepstudy, package = 'lme4')
fm <- lme4::lmer(Reaction ~ Days + (Days | Subject), sleepstudy)

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

## End(Not run)
```

covratio.default *Influence on precision of fixed effects in HLMs*

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

```
## Default S3 method:
covratio(model, ...)

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

## S3 method for class 'mer'
covratio(model, level = 1, delete = NULL, ...)

## S3 method for class 'lmerMod'
covratio(model, level = 1, delete = NULL, ...)

## S3 method for class 'lme'
covratio(model, level = 1, delete = NULL, ...)

## S3 method for class 'mer'
covtrace(model, level = 1, delete = NULL, ...)

## S3 method for class 'lmerMod'
covtrace(model, level = 1, delete = NULL, ...)

## S3 method for class 'lme'
covtrace(model, level = 1, delete = NULL, ...)
```

Arguments

model	fitted model object of class mer or lmerMod
...	do not use
level	variable used to define the group for which cases will be deleted. If level = 1 (default), 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.

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

Christensen, R., Pearson, L., & Johnson, W. (1992) Case-deletion diagnostics for mixed models. *Technometrics*, **34**(1), 38–45.

Schabenberger, O. (2004) Mixed Model Influence Diagnostics, in *Proceedings of the Twenty-Ninth SAS Users Group International Conference*, SAS Users Group International.

See Also

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

Examples

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

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

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

## Not run:
## A larger example
data(Exam, package = 'mlmRev')
fm <- lme4::lmer(normexam ~ standLRT * schavg + (standLRT | school), data = Exam)

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

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

## End(Not run)
```

```
# covtrace for individual observations
ss.ct1 <- covtrace(ss)

# covtrace for subject-level deletion
ss.ct2 <- covtrace(ss, level = "Subject")

## Not run:
## Returning to the larger example
# covtrace for individual observations
ct1 <- covtrace(fm)

# covtrace for school-level deletion
ct2 <- covtrace(fm, level = "school")

## End(Not run)
```

`dotplot_diag` *Dot plots for influence diagnostics*

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

Arguments

`x`	values of the diagnostic of interest
`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.
`data`	data frame to use (optional)

index	optional parameter to specify index (IDs) of x values. If NULL(default), values will be indexed in the order of the vector passed to x.
modify	specifies the geom to be used to produce a space-saving modification: either "dotplot" or "boxplot"
...	other arguments to be passed to ggplot()

Note

The resulting plot uses coord_flip to rotate the plot, so when adding customized axis labels you will need to flip the names between the x and y axes.

Author(s)

Adam Loy <loyad01@gmail.com>

Examples

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

#Observation level deletion and diagnostics
obs.infl <- hlm_influence(fm, level = 1)

dotplot_diag(x = obs.infl$cooksd, cutoff = "internal", name = "cooks.distance", modify = FALSE)

dotplot_diag(x = obs.infl$mdffits, cutoff = "internal", name = "cooks.distance", modify = FALSE)

# Subject level deletion and diagnostics
subject.infl  <- hlm_influence(fm, level = "Subject")

dotplot_diag(x = subject.infl$cooksd, cutoff = "internal",
             name = "cooks.distance", modify = FALSE)

dotplot_diag(x = subject.infl$mdffits, cutoff = "internal", name = "mdffits", modify = "dotplot")
```

---

extract_design *Extracting covariance matrices from lme*

Description

This function extracts the full covariance matrices from a mixed/hierarchical linear model fit using lme.

Usage

```
extract_design(b)
```

Arguments

b	a fitted model object of class lme.

Value

A list of matrices is returned.

- D contains the covariance matrix of the random effects.
- V contains the covariance matrix of the response.
- X contains the fixed-effect model matrix.
- Z contains the random-effect model matrix.

Author(s)

Adam Loy <loyad01@gmail.com>

References

This method has been adapted from the method mgcv::extract.lme.cov in the mgcv package, written by Simon N. Wood <simon.wood@r-project.org>.

---

hlm_augment *Calculating residuals and influence diagnostics for HLMs*

Description

This function is used to compute residuals, fitted values, and influence diagnostics for a hierarchical linear model. The residuals and fitted values are computed using Least Squares(LS) and Empirical Bayes (EB) methods. The influence diagnostics are computed through one step approximations.

Usage

```
hlm_augment(object, ...)

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

## S3 method for class 'lmerMod'
hlm_augment(object, level = 1, include.ls = TRUE, data = NULL, ...)

## S3 method for class 'lme'
hlm_augment(object, level = 1, include.ls = TRUE, ...)
```

Arguments

object	an object of class lmerMod or lme.
...	currently not used
level	which residuals should be extracted and what cases should be deleted for influence diagnostics. If level = 1 (default), then within-group (case-level) residuals are returned and influence diagnostics are calculated for individual observations. Otherwise, level should be the name of a grouping factor as defined in flist for a lmerMod object or as in groups for a lme object. This will return between-group residuals and influence diagnostics calculated for each group.
include.ls	a logical indicating if LS residuals should be included in the return tibble. include.ls = FALSE decreases runtime substantially.
data	the original data frame passed to 'lmer'. This is only necessary for 'lmerMod' models where 'na.action = "na.exclude"'

Details

The hlm_augment function combines functionality from hlm_resid and hlm_influence for a simpler way of obtaining residuals and influence diagnostics. Please see ?hlm_resid and ?hlm_influence for additional information about the returned values.

Note

hlm_augment does not allow for the deletion of specific cases, the specification of other types of leverage, or the use of full refits of the model instead of one step approximations for influence diagnostics. If this additional functionality is desired, hlm_influence should be used instead. The additional parameter standardize is available in hlm_resid; if this are desired, hlm_resid should be used instead.

---

hlm_resid.default *Calculating residuals from HLMs*

Description

hlm_resid takes a hierarchical linear model fit as a lmerMod or lme object and extracts residuals and predicted values from the model, using Least Squares (LS) and Empirical Bayes (EB) methods. It then appends them to the model data frame in the form of a tibble inspired by the augment function in broom. This unified framework enables the analyst to more easily conduct an upward residual analysis during model exploration/checking.

Usage

```
## Default S3 method:
hlm_resid(object, ...)

## S3 method for class 'lmerMod'
hlm_resid(
```

```
  object,
  level = 1,
  standardize = FALSE,
  include.ls = TRUE,
  data = NULL,
  ...
)

## S3 method for class 'lme'
hlm_resid(
  object,
  level = 1,
  standardize = FALSE,
  include.ls = TRUE,
  data = NULL,
  ...
)
```

Arguments

object	an object of class lmerMod or lme.
...	do not use
level	which residuals should be extracted: 1 for within-group (case-level) residuals, the name of a grouping factor for between-group residuals (as defined in flist in lmerMod objects or in groups in lme objects)
standardize	for any level, if standardize = TRUE the standardized residuals will be returned for any group; for level-1 only, if standardize = "semi" then the semi-standardized level-1 residuals will be returned
include.ls	a logical indicating if LS residuals be included in the return tibble. include.ls = FALSE decreases runtime substantially.
data	if na.action = na.exclude, the user must provide the data set used to fit the model, otherwise NULL.

Details

The hlm_resid function provides a wrapper that will extract residuals and predicted values from a fitted lmerMod or lme object. The function provides access to residual quantities already made available by the functions resid, predict, and ranef, but adds additional functionality. Below is a list of types of residuals and predicted values that are extracted and appended to the model data.

level-1 residuals

.resid **and** .fitted Residuals calculated using the EB method (using maximum likelihood). Level-1 EB residuals are interrelated with higher level residuals. Equivalent to the residuals extracted by resid(object) and predict(object) respectively. When standardize = TRUE, residuals are standardized by sigma components of the model object.

.ls.resid **and** .ls.fitted Residuals calculated calculated by fitting separate LS regression models for each group. Level-1 LS residuals are unconfounded by higher level residuals, but unreliable for small within-group sample sizes.

`.mar.resid` **and** `.mar.fitted` Marginal residuals only consider the fixed effect portion of the estimates. Equivalent to `resid(object, level=0)` in `nlme`, not currently implemented within the `lme4::resid` function. When `standardize = TRUE`, Cholesky marginal residuals are returned.

higher-level residuals (random effects)

`.ranef.var_name` The group level random effects using the EB method of estimating parameters. Equivalent to `ranef(object)` on the specified level. EB residuals are preferred at higher levels as LS residuals are dependent on a large sample size.

`.ls.var_name` The group level random effects using the LS method of estimating parameters. Calculated using `ranef` on a `lmList` object to compare the random effects of individual models to the global model.

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

Author(s)

Adam Loy <loyad01@gmail.com>, Jack Moran, Jaylin Lowe

References

Hilden-Minton, J. (1995) Multilevel diagnostics for mixed and hierarchical linear models. University of California Los Angeles.

Houseman, E. A., Ryan, L. M., & Coull, B. A. (2004) Cholesky Residuals for Assessing Normal Errors in a Linear Model With Correlated Outcomes. *Journal of the American Statistical Association*, 99(466), 383–394.

David Robinson and Alex Hayes (2020). broom: Convert Statistical Analysis Objects into Tidy Tibbles. R package version 0.5.6. https://CRAN.R-project.org/package=broom

See Also

`hlm_augment`, `resid`, `lme4::ranef`

Examples

```
data(sleepstudy, package = "lme4")
fm.lmer <- lme4::lmer(Reaction ~ Days + (Days|Subject), sleepstudy)
fm.lme <- nlme::lme(Reaction ~ Days, random = ~Days|Subject, sleepstudy)

# level-1 and marginal residuals
fm.lmer.res1 <- hlm_resid(fm.lmer) ## raw level-1 and mar resids
fm.lmer.res1
fm.lme.std1 <- hlm_resid(fm.lme, standardize = TRUE) ## std level-1 and mar resids
fm.lme.std1

# level-2 residuals
fm.lmer.res2 <- hlm_resid(fm.lmer, level = "Subject") ## level-2 ranefs
fm.lmer.res2
fm.lme.res2 <- hlm_resid(fm.lme, level = "Subject", include.ls = FALSE) ##level-2 ranef, no LS
fm.lme.res2
```

HLMdiag *Diagnostic tools for hierarchical (multilevel) linear models*

Description

HLMdiag provides a suite of diagnostic tools for hierarchical (multilevel) linear models fit using the lme4 or nlme packages. These tools are grouped below by purpose. See the help documentation for additional information about each function.

Details

Residual analysis

HLMdiag's hlm_resid function provides a wrapper that extracts residuals and fitted values for individual observations or groups of observations. In addition to being a wrapper function for functions implemented in the lme4 and nlme packages, hlm_resid provides access to the marginal and least squares residuals.

Influence analysis

HLMdiag's hlm_influence function provides a convenient wrapper to obtain influence diagnostics for each observation or group of observations appended to the data used to fit the model. The diagnostics returned by hlm_influence include Cook's distance, MDFFITS, covariance trace (covtrace), covariance ratio (covratio), leverage, and relative variance change (RVC). HLMdiag also contains functions to calculate these diagnostics individually, as discussed below.

Influence on fitted values

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

Influence on precision of fixed effects

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

Influence on variance components

HLMdiag's 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 ggplot2 framework by providing dotplots for influence diagnostics (dotplot_diag), grouped Q-Q plots (group_qqnorm), and Q-Q plots that combine the functionality of qqnorm and qqline (ggplot_qqnorm).

Author(s)

Maintainer: Adam Loy <loyad01@gmail.com>

Authors:

- Jaylin Lowe
- Jack Moran

See Also

Useful links:

- https://github.com/aloy/HLMdiag
- Report bugs at https://github.com/aloy/HLMdiag/issues

`HLMdiag-defunct` *Defunct functions in package HLMdiag*

Description

These functions are defunct and no longer available.

Usage

```
HLMresid(...)

HLMresid.default(...)

HLMresid.lmerMod(...)

HLMresid.mer(...)

diagnostics(...)

group_qqnorm(...)

ggplot_qqnorm(...)
```

Arguments

`...`	arguments passed to defunct functions

Details

`HLMresid` is replaced by `hlm_resid`

`diagnostics` is replaced by `hlm_influence`

`group_qqnorm` and `group_qqnorm` are replaced by functions in **qqplotr**. See `stat_qq_point`, `stat_qq_line`, and `stat_qq_band`.

HLMdiag-deprecated *Deprecated functions in HLMdiag*

Description

These functions still work but will be removed (defunct) in the next version.

Details

- HLMresid: This function is deprecated, and will be removed in the next version of this package.
- diagnostics: This function is deprecated, and will be removed in the next version of this package.

leverage.default *Leverage for HLMs*

Description

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

Usage

```
## Default S3 method:
leverage(model, ...)

## S3 method for class 'mer'
leverage(model, level = 1, ...)

## S3 method for class 'lmerMod'
leverage(model, level = 1, ...)

## S3 method for class 'lme'
leverage(model, level = 1, ...)
```

Arguments

model	fitted model object of class mer of lmerMod
...	do not use
level	the level at which the leverage should be calculated: either 1 for observation level leverage (default) 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.

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

$$H_2^* = ZDZ'$$

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

Demidenko, E., & Stukel, T. A. (2005) Influence analysis for linear mixed-effects models. *Statistics in Medicine*, **24**(6), 893–909.

Nobre, J. S., & Singer, J. M. (2011) Leverage analysis for linear mixed models. *Journal of Applied Statistics*, **38**(5), 1063–1072.

See Also

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

Examples

```
data(sleepstudy, package = 'lme4')
fm <- lme4::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)
```

`LSresids.default` *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

```
## 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, standardize = FALSE, ...)

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

Arguments

`object`	an object of class `mer` or `lmerMod`.
`...`	do not use
`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. Removed in version 0.3.2.
`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.

Author(s)

Adam Loy <loyad01@gmail.com>

References

Hilden-Minton, J. (1995) Multilevel diagnostics for mixed and hierarchical linear models. University of California Los Angeles.

See Also

HLMresid

`mdffits.default` *Influence on fixed effects of HLMs*

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

```
## Default S3 method:
mdffits(model, ...)

## S3 method for class 'mer'
cooks.distance(model, level = 1, delete = NULL, ...)

## S3 method for class 'lmerMod'
cooks.distance(model, level = 1, delete = NULL, include.attr = FALSE, ...)

## S3 method for class 'lme'
cooks.distance(model, level = 1, delete = NULL, include.attr = FALSE, ...)

## S3 method for class 'mer'
mdffits(object, level = 1, delete = NULL, ...)

## S3 method for class 'lmerMod'
mdffits(model, level = 1, delete = NULL, include.attr = FALSE, ...)

## S3 method for class 'lme'
mdffits(model, level = 1, delete = NULL, include.attr = FALSE, ...)
```

Arguments

model	fitted model of class mer or lmerMod
...	do not use
level	variable used to define the group for which cases will be deleted. If level = 1 (default), 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.
include.attr	logical value determining whether the difference between the full and deleted parameter estimates should be included. If FALSE (default), a numeric vector of Cook's distance or MDFFITS is returned. If TRUE, a tibble with the Cook's distance or MDFFITS values in the first column and the parameter differences in the remaining columns is returned.
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) as the default. If include.attr = TRUE, then a tibble is returned. The first column consists of the Cook's distance or MDFFITS values, and the later columns capture 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

Christensen, R., Pearson, L., & Johnson, W. (1992) Case-deletion diagnostics for mixed models. *Technometrics*, **34**, 38–45.

Schabenberger, O. (2004) Mixed Model Influence Diagnostics, in *Proceedings of the Twenty-Ninth SAS Users Group International Conference*, SAS Users Group International.

See Also

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

Examples

```
data(sleepstudy, package = 'lme4')
ss <- lme4::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, level = "Subject")

## Not run:
data(Exam, package = 'mlmRev')
fm <- lme4::lmer(normexam ~ standLRT * schavg + (standLRT | school), Exam)

# Cook's distance for individual observations
cd.lev1 <- cooks.distance(fm)

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

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


## End(Not run)


# MDFFITS  for individual observations
ss.m1 <- mdffits(ss)

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

## Not run:

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

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

## End(Not run)
```

---

pull_resid.default *Computationally Efficient HLM Residuals*

Description

pull_resid takes a hierarchical linear model fit as a lmerMod or lme object and returns various types of level-1 residuals as a vector. Because the pull_resid only calculates one type of residual, it is more efficient than using hlm_resid and indexing the resulting tibble. pull_resid is designed to be used with methods that take a long time to run, such as the resampling methods found in the lmeresampler package.

Usage

```
## Default S3 method:
pull_resid(object, ...)

## S3 method for class 'lmerMod'
pull_resid(object, type = "ls", standardize = FALSE, ...)

## S3 method for class 'lme'
pull_resid(object, type = "ls", standardize = FALSE, ...)
```

Arguments

object	an object of class lmerMod or lme.
...	not in use
type	which residuals should be returned. Can be either 'ls', 'eb', or 'marginal'
standardize	a logical indicating if residuals should be standardized

Details

type = "ls" Residuals calculated by fitting separate LS regression models for each group. LS residuals are unconfounded by higher level residuals, but unreliable for small within-group sample sizes. When standardize = TRUE, residuals are standardized by sigma components of the model object.

type = "eb" Residuals calculated using the empirical Bayes (EB) method using maximum likelihood. EB residuals are interrelated with higher level residuals. When standardize = TRUE, residuals are standardized by sigma components of the model object.

type = "marginal" Marginal residuals only consider the fixed effect portion of the estimates. When standardize = TRUE, Cholesky residuals are returned.

See Also

hlm_resid

radon *Radon data*

Description

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

Usage

```
data(radon)
```

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.

References

Price, P. N., Nero, A. V. and Gelman, A. (1996) Bayesian prediction of mean indoor radon concentrations for Minnesota counties. *Health Physics*. **71**(6), 922–936.

Gelman, A. and Hill, J. (2007) *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press.

resid_conditional.default *Conditional residuals*

Description

Calculates conditional residuals of `lmerMod` and `lme` model objects.

Usage

```
## Default S3 method:
resid_conditional(object, type)

## S3 method for class 'lmerMod'
resid_conditional(
  object,
  type = c("raw", "pearson", "studentized", "cholesky")
)

## S3 method for class 'lme'
resid_conditional(
  object,
  type = c("raw", "pearson", "studentized", "cholesky")
)
```

Arguments

`object`	an object of class `lmerMod` or `lme`.
`type`	a character string specifying what type of residuals should be calculated. It is set to `"raw"` (observed - fitted) by default. Other options include `"pearson"`, `"studentized"`, and `"cholesky"`. Partial matching of arguments is used, so only the first character needs to be provided.

Details

For a model of the form $Y = X\beta + Zb + \epsilon$, four types of marginal residuals can be calculated:

`raw` $e = Y - X\hat{beta} - Z\hat{b}$

`pearson` $e/\sqrt{diag(\hat{Var}(Y|b))}$

`studentized` $e/\sqrt{diag(\hat{Var}(e))}$

`cholesky` $\hat{C}^{-1}e$ where $\hat{C}\hat{C}' = \hat{Var}(e)$

Value

A vector of conditional residuals.

References

Singer, J. M., Rocha, F. M. M., & Nobre, J. S. (2017). Graphical Tools for Detecting Departures from Linear Mixed Model Assumptions and Some Remedial Measures. *International Statistical Review*, **85**, 290–324.

Schabenberger, O. (2004) Mixed Model Influence Diagnostics, in *Proceedings of the Twenty-Ninth SAS Users Group International Conference*, SAS Users Group International.

`resid_marginal.default` *Marginal residuals*

Description

Calculates marginal residuals of `lmerMod` and `lme` model objects.

Usage

```
## Default S3 method:
resid_marginal(object, type)

## S3 method for class 'lmerMod'
resid_marginal(object, type = c("raw", "pearson", "studentized", "cholesky"))

## S3 method for class 'lme'
resid_marginal(object, type = c("raw", "pearson", "studentized", "cholesky"))
```

Arguments

`object`	an object of class `lmerMod` or `lme`.
`type`	a character string specifying what type of residuals should be calculated. It is set to `"raw"` (observed - fitted) by default. Other options include `"pearson"`, `"studentized"`, and `"cholesky"`. Partial matching of arguments is used, so only the first character needs to be provided.

Details

For a model of the form $Y = X\beta + Zb + \epsilon$, four types of marginal residuals can be calculated:

`raw` $r = Y - X\hat{beta}$

`pearson` $r/\sqrt{diag(\hat{Var}(Y))}$

`studentized` $r/\sqrt{diag(\hat{Var}(r))}$

`cholesky` $\hat{C}^{-1}r$ where $\hat{C}\hat{C}' = \hat{Var}(Y)$

Value

A vector of marginal residuals.

References

Singer, J. M., Rocha, F. M. M., & Nobre, J. S. (2017). Graphical Tools for Detecting Departures from Linear Mixed Model Assumptions and Some Remedial Measures. *International Statistical Review*, **85**, 290–324.

Schabenberger, O. (2004) Mixed Model Influence Diagnostics, in *Proceedings of the Twenty-Ninth SAS Users Group International Conference*, SAS Users Group International.

`resid_ranef` *Random effects residuals*

Description

Calculates Random effects residuals of `lmerMod` model objects.

Usage

```
resid_ranef(object, level, which, standardize)
```

Arguments

`object`	an object of class `lmerMod`.
`level`	DESCRIPTION
`which`	DESCRIPTION
`standardize`	DESCRIPTION

Value

A vector of conditional residuals.

`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

```
## 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, ...)

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

Arguments

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

Author(s)

Adam Loy <loyad01@gmail.com>

References

Loy, A. & Hofmann, H. (in press). Are you Normal? The Problem of Confounded Residual Structures in Hierarchical Linear Models. *Journal of Computational and Graphical Statistics*.

---

rvc.default *Relative variance change for HLMs*

Description

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

Usage

```
## Default S3 method:
rvc(model, ...)

## S3 method for class 'mer'
rvc(model, level = 1, delete = NULL, ...)
```

```
## S3 method for class 'lmerMod'
rvc(model, level = 1, delete = NULL, ...)

## S3 method for class 'lme'
rvc(model, level = 1, delete = NULL, ...)
```

Arguments

model	fitted model object of class mer or lmerMod
...	do not use
level	variable used to define the group for which cases will be deleted. If level = 1 (default), 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.

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

Dillane, D. (2005) Deletion Diagnostics for the Linear Mixed Model. Ph.D. thesis, Trinity College Dublin

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 <- lme4::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.

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

uerate local area unemployment rate for that year

ue.7

ue.centert1

ue.mean

ue.person.cen

ue1

Source

These data are originally from the 1979 National Longitudinal Survey on Youth (NLSY79).

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: https://stats.oarc.ucla.edu/other/examples/alda/

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

References

Singer, J. D. and Willett, J. B. (2003), *Applied Longitudinal Data Analysis: Modeling Change and Event Occurrence*, New York: Oxford University Press.

Cook, D. and Swayne, D. F. (2007), *Interactive and Dynamic Graphics for Data Analysis with R and GGobi*, Springer.

Examples

```
str(wages)
summary(wages)

## Not run:
library(lme4)
lmer(lnw ~ exper + (exper | id), data = wages)

## End(Not run)
```

Index

∗ **datasets**
ahd, 4
autism, 4
radon, 28
wages, 34
∗ **hplot**
dotplot_diag, 13
∗ **models**
adjust_lmList.formula, 3
case_delete.default, 5
covratio.default, 11
hlm_resid.default, 16
leverage.default, 21
LSresids.default, 23
mdffits.default, 24
rvc.default, 32
varcomp.mer, 34
∗ **package**
HLMdiag, 19
∗ **regression**
adjust_lmList.formula, 3
case_delete.default, 5
covratio.default, 11
hlm_resid.default, 16
leverage.default, 21
LSresids.default, 23
mdffits.default, 24
rvc.default, 32
varcomp.mer, 34

adjust_lmList *(*adjust_lmList.formula*)*, 3
adjust_lmList,formula,data.frame-method *(*adjust_lmList.formula*)*, 3
adjust_lmList.formula, 3
ahd, 4
autism, 4

case_delete *(*case_delete.default*)*, 5
case_delete.default, 5
compare_eb_ls, 8
cooks.distance, *19*
cooks.distance *(*mdffits.default*)*, 24
cooks.distance.case_delete, 8
cooks.distance.mer, *12*, *22*, *33*
covratio, *19*
covratio *(*covratio.default*)*, 11
covratio.case_delete *(*cooks.distance.case_delete*)*, 8
covratio.default, 11
covratio.mer, *22*, *26*, *33*
covtrace, *19*
covtrace *(*covratio.default*)*, 11
covtrace.case_delete *(*cooks.distance.case_delete*)*, 8
covtrace.mer, *22*, *26*, *33*

diagnostics, *21*
diagnostics *(*HLMdiag-defunct*)*, 20
dotplot_diag, 13, *19*

extract_design, 14

ggplot_qqnorm, *19*
ggplot_qqnorm *(*HLMdiag-defunct*)*, 20
group_qqnorm, *19*
group_qqnorm *(*HLMdiag-defunct*)*, 20

hlm_augment, 15, *18*
hlm_influence, *19*, *20*
hlm_influence *(*cooks.distance.case_delete*)*, 8
hlm_resid, *19*, *20*, *27*
hlm_resid *(*hlm_resid.default*)*, 16
hlm_resid.default, 16
HLMdiag, 19
HLMdiag-defunct, 20
HLMdiag-deprecated, 21
HLMdiag-package *(*HLMdiag*)*, 19
HLMresid, *21*, *24*
HLMresid *(*HLMdiag-defunct*)*, 20

leverage, *19*
leverage *(*leverage.default*)*, 21
leverage.default, 21
leverage.mer, *12*, *26*, *33*
lm, *3*
lme4::ranef, *18*
lmList, *3*
LSresids *(*LSresids.default*)*, 23
LSresids.default, 23

mdffits, *19*
mdffits *(*mdffits.default*)*, 24
mdffits.case_delete
*(*cooks.distance.case_delete*)*, 8
mdffits.default, 24
mdffits.mer, *12*, *22*, *33*

package-HLMdiag *(*HLMdiag*)*, 19
pull_resid *(*pull_resid.default*)*, 26
pull_resid.default, 26

qqline, *19*
qqnorm, *19*

radon, 28
resid, *18*
resid_conditional
*(*resid_conditional.default*)*, 28
resid_conditional.default, 28
resid_marginal
*(*resid_marginal.default*)*, 30
resid_marginal.default, 30
resid_ranef, 31
rotate_ranef *(*rotate_ranef.default*)*, 31
rotate_ranef.default, 31
rvc, *19*
rvc *(*rvc.default*)*, 32
rvc.case_delete
*(*cooks.distance.case_delete*)*, 8
rvc.default, 32
rvc.mer, *12*, *22*, *26*

stat_qq_band, *20*
stat_qq_line, *20*
stat_qq_point, *20*

varcomp.mer, 34

wages, 34