Package ‘merDeriv’

October 13, 2022

Title Case-Wise and Cluster-Wise Derivatives for Mixed Effects Models

Version 0.2-4

Description
Compute case-wise and cluster-wise derivative for mixed effects models with respect to fixed effects parameter, random effect (co)variances, and residual variance. This material is partially based on work supported by the National Science Foundation under Grant Number 1460719.

Depends R (>= 3.2.3), lme4 (>= 1.1-10), stats, methods, nonnest2,
sandwich, lavaan

Imports utils, Matrix, numDeriv

License GPL (>= 2)

URL https://github.com/nctingwang/merDeriv

NeedsCompilation no

Maintainer Ting Wang <szpku.grady@gmail.com>

Suggests tinytest, mirt, lmeInfo

Author Ting Wang [aut, cre],
Edgar Merkle [aut] (<https://orcid.org/0000-0001-7158-0653>),
Yves Rosseel [ctb]

Repository CRAN

Date/Publication 2022-03-11 23:30:05 UTC

R topics documented:

bread.glmerMod 2
bread.lmerMod 3
estfun.glmerMod 4
estfun.lmerMod 5
llcont.glmerMod 6
llcont.lmerMod 7
vcov.glmerMod 8
vcov.lmerMod 9

Index **11**

`bread.glmerMod` *Extract Bread Component for Huber-White Sandwich Estimator of Generalized Linear Mixed Effects Models*

Description

This function calculates the bread component of the Huber-White sandwich estimator (variance covariance matrix multiplied by the number of clusters) for a generalized linear mixed effects model of class `glmerMod`.

Usage

```
## S3 method for class 'glmerMod'
bread(x, ...)
```

Arguments

`x` An object of class `glmerMod`.

`...` additional arguments, including `full` and `ranpar` (`full` = `FALSE`, `ranpar` = `"var"`; see details).

Value

A *p* by *p* "bread" matrix for the Huber-White sandwich estimator (variance-covariance matrix based on observed Fisher information multiplied by the number of clusters), where *p* represents the number of parameters. If `full` = `FALSE`, returns the variance-covariance matrix of only fixed effect parameters. If `full` = `TRUE` , returns the variance-covariance matrix for all fitted parameters (including fixed effect parameters, random effect (co)variances, and residual variance. If `ranpar` = `"var"`, the random effects are parameterized as variance/covariance; If `ranpar` = `"sd"`, the random effects are parameterized as standard deviation/correlation; If `ranpar` = `"theta"`, the random effects are parameterized as components of Cholesky decomposition.

References

Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software*, **67**(1), 1-48. doi: 10.18637/jss.v067.i01.

Zeileis, A. (2006). Object-Oriented Computation of Sandwich Estimators. Journal of Statistical Software, **16**(9), 1-16. `https://www.jstatsoft.org/v16/i09/`

Examples

```
## Not run:
# The cbpp example
data(finance, package = "smdata")

lme4fit <- glmer(corr ~ jmeth + (1 | item), data = finance,
                 family = binomial, nAGQ = 20)
```

```
# bread component for all parameters
bread(lme4fit, full = TRUE, ranpar = "var")

## End(Not run)
```

bread.lmerMod *Extract Bread Component for Huber-White Sandwich Estimator of Linear Mixed Effects Models*

Description

This function calculates the bread component of the Huber-White sandwich estimator (variance covariance matrix multiplied by the number of clusters) for a linear mixed effects model of class `lmerMod`.

Usage

```
## S3 method for class 'lmerMod'
bread(x, ...)
```

Arguments

`x` An object of class `lmerMod`.

`...` additional arguments, including `full` `information`, and `ranpar` (`full` = `FALSE`, `information` = `"expected"` and `ranpar` = `"var"` are default; see details).

Value

A p by p "bread" matrix for the Huber-White sandwich estimator (variance-covariance matrix multiplied by the number of clusters), where p represents the number of parameters. If `full` = `FALSE`, returns the variance-covariance matrix of only fixed effect parameters. If `full` = `TRUE` , returns the variance-covariance matrix for all fitted parameters (including fixed effect parameters, random effect (co)variances, and residual variance. If `information` = `"expected"`, the variance-covariance matrix is based on the inversion of Fisher information matrix. If `information` = `"observed"`, the variance-covariance matrix is based on the observed Fisher information, which is the negative of Hessian matrix. If `ranpar` = `"var"`, the random effects are parameterized as variance/covariance; If `ranpar` = `"sd"`, the random effects are parameterized as standard deviation/correlation.

References

Wang, T. & Merkle, E. C. (2018). Derivative Computations and Robust Standard Errors for Linear Mixed Effects Models in lme4. *Journal of Statistical Software*, **87**(1), 1-16. doi: 10.18637/jss.v087.c01

Zeileis, A. (2006). Object-Oriented Computation of Sandwich Estimators. Journal of Statistical Software, **16**(9), 1-16. `https://www.jstatsoft.org/v16/i09/`

Examples

```
## Not run:
# The sleepstudy example
lme4fit <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy, REML = FALSE)

# bread component for all parameters
bread(lme4fit, full = TRUE, information = "expected", ranpar = "var")

## End(Not run)
```

`estfun.glmerMod` *Extract Cluster-wise Derivatives for Generalized Linear Mixed Effects Models*

Description

A function for extracting the cluster-wise derivatives of a generalized linear mixed effects models fitted via **lme4**. This function returns the cluster-wise scores, evaluated at the ML estimates.

Usage

```
## S3 method for class 'glmerMod'
estfun(x, ...)
```

Arguments

x	An object of class `glmerMod`.
...	Additional arguments, including `ranpar` (`ranpar = "var"` is default; see details).

Value

A g by p score matrix, corresponding to g clusters and p parameters. For models with multiple clustering variables (three-level models, crossed random effects), an error is thrown. If `ranpar = "var"`, the random effects are parameterized as variance/covariance; If `ranpar = "sd"`, the random effects are parameterized as standard deviation/correlation; If `ranpar = "theta"`, the random effects are parameterized as components of Cholesky decomposition.

References

Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software*, **67**(1), 1-48. doi: 10.18637/jss.v067.i01.

Examples

```
## Not run:
data(finance, package = "smdata")

lme4fit <- glmer(corr ~ jmeth + (1 | item), data = finance,
                 family = binomial, nAGQ = 20)

# clusterwise scores
estfun(lme4fit, ranpar = "var")

## End(Not run)
```

`estfun.lmerMod` *Extract Case-wise and Cluster-wise Derivatives for Linear Mixed Effects Models*

Description

A function for extracting the case-wise and cluster-wise derivatives of a linear mixed effects models fitted via **lme4**. This function returns the case-wise and cluster-wise scores, evaluated at the ML estimates.

Usage

```
## S3 method for class 'lmerMod'
estfun(x, ...)
```

Arguments

`x` An object of class `lmerMod`.

`...` additional arguments, including `level` and `ranpar` (`level = 2` and `ranpar = "var"` are default; see details).

Value

If `level = 2`, a g by p score matrix, corresponding to g clusters and p parameters. If `level = 1`, a n by p score matrix, corresponding to n observations and p parameters. For models with multiple clustering variables (three-level models, crossed random effects), an error is thrown if `level = 2`. If `ranpar = "var"`, the random effects are parameterized as variance/covariance; If `ranpar = "sd"`, the random effects are parameterized as standard deviation/correlation.

References

Wang, T. & Merkle, E. C. (2018). Derivative Computations and Robust Standard Errors for Linear Mixed Effects Models in lme4. *Journal of Statistical Software*, **87**(1), 1-16. doi: 10.18637/jss.v087.c01

Examples

```
## Not run:
# The sleepstudy example
lme4fit <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy, REML = FALSE)

# casewise scores
estfun(lme4fit, level = 1, ranpar = "var")

# clusterwise scores
estfun(lme4fit, level = 2, ranpar = "sd")

## End(Not run)
```

`llcont.glmerMod` *Extract Cluster-wise Log Likelihoods for Generalized Linear Mixed Effects Models*

Description

A function for extracting the cluster-wise log likelihoods of a generalized linear mixed effects model fitted via **lme4**. This function returns the cluster-wise log likelihoods, evaluated at the ML estimates.

Usage

```
## S3 method for class 'glmerMod'
llcont(x, ...)
```

Arguments

`x`	An object of class `glmerMod`.
`...`	Additional arguments.

Value

A vector of log-likelihoods whose length is the number of clusters.

References

Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software*, **67**(1), 1-48. doi: 10.18637/jss.v067.i01.

Merkle, E. C., You, D. and Preacher, K. J., 2016. Testing Nonnested Structural Equation Models. *Psychological Methods*, **21**(2), 151. `https://arxiv.org/pdf/1402.6720v3`

Examples

```
## Not run:
data(finance, package="smdata")

lme4fit <- glmer(corr ~ jmeth + (1 | item), data = finance,
                 family = binomial, nAGQ = 20)

# clusterwise log likelihood
llcont(lme4fit)

## End(Not run)
```

---

llcont.lmerMod *Extract Case-wise Log Likelihoods for Linear Mixed Effects Models*

Description

A function for extracting the case-wise log likelihoods of a linear mixed effects model fitted via **lme4**. This function returns the case-wise log likelihoods, evaluated at the ML estimates.

Usage

```
## S3 method for class 'lmerMod'
llcont(x, ...)
```

Arguments

x	An object of class lmerMod.
...	additional arguments, including level (level = 2 is default; see details).

Value

If level = 2, a vector of log-likelihoods whose length is the number of clusters. If level = 1, a vector of length *n*, containing log-likelihoods for all *n* observations.

References

Wang, T. & Merkle, E. C. (2018). Derivative Computations and Robust Standard Errors for Linear Mixed Effects Models in lme4. *Journal of Statistical Software*, **87**(1), 1-16. doi: 10.18637/jss.v087.c01

Merkle, E. C., You, D. and Preacher, K. J., 2016. Testing Nonnested Structural Equation Models. *Psychological Methods*, **21**(2), 151. https://arxiv.org/pdf/1402.6720v3

Examples

```
## Not run:
# The sleepstudy example
lme4fit <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy, REML = FALSE)

# clusterwise log likelihood
llcont(lme4fit)

## End(Not run)
```

`vcov.glmerMod` *Extract Variance-Covariance Matrix of all Parameters for Generalized Linear Mixed Effects Models*

Description

This function calculates the variance-covariance matrix for all parameters (fixed and random effect) in a generalized linear mixed effects model of class `glmerMod`.

Usage

```
## S3 method for class 'glmerMod'
vcov(object, ...)
```

Arguments

`object`	An object of class `glmerMod`.
`...`	additional arguments, including `full` and `ranpar` (`full = FALSE` and `ranpar = "var"` are default; see details).

Value

A p by p variance-covariance matrix, where p represents the number of parameters. If `full = FALSE`, returns the variance-covariance matrix of only fixed effect parameters. If `full = TRUE` , returns the variance-covariance matrix for all fitted parameters (including fixed effect parameters and random effect (co)variances). The variance-covariance matrix is based on the negative of Hessian matrix, which is extracted from `lme4`. If `ranpar = "var"`, the random effects are parameterized as variance/covariance; If `ranpar = "sd"`, the random effects are parameterized as standard deviation/correlation; If `ranpar = "theta"`, the random effects are parameterized as components of Cholesky decomposition.

References

Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software*, **67**(1), 1-48. doi: 10.18637/jss.v067.i01.

Examples

```
## Not run:
# The cbpp example
data(finance, package="smdata")

lme4fit <- glmer(corr ~ jmeth + (1 | item), data = finance,
                 family = binomial, nAGQ = 20)

# variance covariance matrix for all parameters
vcov(lme4fit, full = TRUE, ranpar = "var")

## End(Not run)
```

`vcov.lmerMod` *Extract Variance-Covariance Matrix of all Parameters for Linear Mixed Effects Models*

Description

This function calculates the variance-covariance matrix for all parameters (fixed, random effect, and residual) in a linear mixed effects model of class `lmerMod`.

Usage

```
## S3 method for class 'lmerMod'
vcov(object, ...)
```

Arguments

`object` An object of class `lmerMod`.

`...` additional arguments, including `full`, `information` and `ranpar` (`full = FALSE`, `information = "expected"` and `ranpar = "var"` are default; see details).

Value

A *p* by *p* variance-covariance matrix, where *p* represents the number of parameters. If `full = FALSE`, returns the variance-covariance matrix of only fixed effect parameters. If `full = TRUE` , returns the variance-covariance matrix for all fitted parameters (including fixed effect parameters, random effect (co)variances, and residual variance. If `information = "expected"`, the variance-covariance matrix is based on the inversion of Fisher information matrix. If `information = "observed"`, the variance-covariance matrix is based on the observed Fisher information, which is the negative of Hessian matrix. If `ranpar = "var"`, the random effects are parameterized as variance/covariance; If `ranpar = "sd"`, the random effects are parameterized as standard deviation/correlation.

References

Wang, T. & Merkle, E. C. (2018). Derivative Computations and Robust Standard Errors for Linear Mixed Effects Models in lme4. *Journal of Statistical Software*, **87**(1), 1-16. doi: 10.18637/jss.v087.c01

Examples

```
## Not run:
# The sleepstudy example
lme4fit <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy, REML = FALSE)

# variance covariance matrix for all parameters
vcov(lme4fit, full = TRUE, ranpar = "var")

## End(Not run)
```

Index

bread.glmerMod, 2
bread.lmerMod, 3

estfun.glmerMod, 4
estfun.lmerMod, 5

glmerMod, *2*, *4*, *6*, *8*

llcont.glmerMod, 6
llcont.lmerMod, 7
lmerMod, *3*, *5*, *7*, *9*

vcov.glmerMod, 8
vcov.lmerMod, 9