Package: nlmm (via r-universe)

September 11, 2024

Type Package

Title Generalized Laplace Mixed-Effects Models

Version 1.1.0

Date 2023-09-04

Maintainer Marco Geraci <marco.geraci@uniroma1.it>

Depends R (>= 3.0.0), nlme

Imports gsl, lqmm, MASS, Matrix, mvtnorm, numDeriv, statmod, stats, Qtools, Rcpp (>= 0.12.13), utils

LinkingTo Rcpp, RcppArmadillo, BH

Description Provides functions to fit linear mixed models based on convolutions of the generalized Laplace (GL) distribution. The GL mixed-effects model includes four special cases with normal random effects and normal errors (NN), normal random effects and Laplace errors (NL), Laplace random effects and normal errors (LN), and Laplace random effects and Laplace errors (LL). The methods are described in Geraci and Farcomeni (2020, Statistical Methods in Medical Research) <doi:10.1177/0962280220903763>.

License GPL (>= 2)

LazyLoad yes

Repository https://marco-geraci.r-universe.dev

RemoteUrl https://github.com/marco-geraci/nlmm

RemoteRef HEAD

RemoteSha 2c2767b7c5bca656bb2e00e4ce80b69f93c85a67

Contents

`nlmm-package` *nlmm: Generalized Laplace Mixed-Effects Models*

Description

The nlmm package provides functions to fit linear mixed models based on convolutions of the generalized Laplace (GL) distribution. The GL mixed-effects model includes four special cases with normal random effects and normal errors (NN), normal random effects and Laplace errors (NL), Laplace random effects and normal errors (LN), and Laplace random effects and Laplace errors (LL). The methods are described in Geraci and Farcomeni (2020). See also Geraci (2017) for details on special cases.

Details

Package:	nlmm
Type:	Package
Version:	1.1.0
Date:	2023-09-04
License:	GPL (>=3)
LazyLoad:	yes

Author(s)

Marco Geraci [aut, cph, cre], Alessio Farcomeni [ctb]

Maintainer: Marco Geraci <marco.geraci@uniroma1.it>

References

Geraci M (2017). Mixed-effects models using the normal and the Laplace distributions: A 2 x 2 convolution scheme for applied research. arXiv:1712.07216v1 [stat.ME]. URL: https://arxiv.org/abs/1712.07216v1.

Geraci, M. and Farcomeni A (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research 29(9), 2665-2682.

`fixef.nlmm` *Extract Generalized Mixed-Effects Models Coefficients*

Description

`fixef` extracts estimated fixed effects from `nlmm` objects.

Usage

```
## S3 method for class 'nlmm'
fixef(object, ...)
```

Arguments

`object`	a `nlmm` object.
`...`	not used.

Value

a vector of estimated fixed effects.

Author(s)

Marco Geraci

See Also

`nlmm` `summary.nlmm`

Examples

```
## Not run:
data(rats)
fit <- nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats,
control = nlmmControl(multistart = FALSE))
fixef(fit)

## End(Not run)
```

`generate.data` *Simulate Data from Mixed-Effects Models*

Description

This function generates data from a 2-level hierarchical design.

Usage

```
generate.data(R, n, M, sigma_1 = NULL, sigma_2 = NULL,
 shape_1 = NULL, shape_2 = NULL, dist.u, dist.e,
 beta, gamma, fixed = FALSE, seed = round(runif(1,1,1000)))
```

Arguments

`R`	number of replications.
`n`	number of observations within cluster.
`M`	number of clusters.
`sigma_1`	scale parameter for the random effects.
`sigma_2`	scale parameter for the errors.
`shape_1`	shape parameter for the random effects.
`shape_2`	shape parameter for the errors.
`dist.u`	distribution of the random effects.
`dist.e`	distribution of the errors.
`beta`	vector of coefficients for fixed effects.
`gamma`	vector of coefficients for heteroscedasticity.
`fixed`	logical flag. See details.
`seed`	seed for random number generation.

Details

This function generates data as in the simulation study by Geraci and Farcomeni (2020). The data-generating model is

$$y[ij] = \beta[0] + \beta[1]x[ij] + \beta[2]z[ij] + u[i] + v[i]x[ij] + (\gamma[0] + \gamma[1]x[ij])e[ij]$$

where $(u[i], v[i])$ follows a distribution with scale `sigma_1` and shape `shape_1`, and e follows a distribution with scale `sigma_2` and shape `shape_2`.

The scale parameter `sigma_1` must be a 1 by 1 or a 2 by 2 matrix. In the former case, the model will include only random intercepts. In the latter case, then both random intercepts and slopes will be included. Currently, no more than 2 random effects can be specified. The scale parameter `sigma_2` must be a matrix `n` by `n`.

The options for dist.u and dist.e are: multivariate normal ("norm") (rmvnorm), multivariate symmetric Laplace ("laplace") (rmal), multivariate symmetric generalized Laplace ("genlaplace") rmgl, and multivariate Student's t ("t") (rmvt).

The shape parameter specifies the degrees of freedom for Student's t and chi-squared, and the kurtosis of the generalized Laplace.

The values $x[ij]$ are generated as $x[ij] = \delta[i] + \zeta[ij]$, where $\delta[i]$ and $\zeta[ij]$ are independent standard normal. If the argument fixed = TRUE, then $x[ij] = j$. The values $z[ij]$ are generated from Bernoullis with probability 0.5.

Value

nlmm returns an object of class nlmm.

The function summary is used to obtain and print a summary of the results.

An object of class nlmm is a list containing the following components:

Y	a matrix RxN, where $N = nxM$, with responses
X	an array $Nx3xR$ with fixed design matrix
group	vector of length N with cluster labels
u	an array $Mx2xR$ with random effects
e	a matrix RxN with errors

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

See Also

nlmm

Examples

```
# Simulate 10 replications from a homoscedastic normal mixed model.
generate.data(R = 10, n = 3, M = 5, sigma_1 = diag(2), sigma_2 = diag(3),
shape_1 = NULL, shape_2 = NULL, dist.u = "norm", dist.e = "norm",
beta = c(1,2,1), gamma = c(1,0))

# Simulate 10 replications from a generalized Laplace. Note: the shape
# parameter that is passed to rmgl corresponds to the reciprocal of the
# parameter alpha in Geraci and Farcomeni (2020)
generate.data(R = 10, n = 3, M = 5, sigma_1 = diag(2), sigma_2 = diag(3),
shape_1 = 1/0.5, shape_2 = 1/0.5, dist.u = "genlaplace", dist.e = "genlaplace",
beta = c(1,2,1), gamma = c(1,0))
```

GenLaplace *The Univariate Symmetric Generalized Laplace Distribution*

Description

Density, distribution function, quantile function and random generation for the univariate symmetric generalized Laplace distribution.

Usage

```
dgl(x, sigma = 1, shape = 1, log = FALSE)
pgl(x, sigma = 1, shape = 1, lower.tail = TRUE, log.p = FALSE)
qgl(p, sigma = 1, shape = 1, lower.tail = TRUE, log.p = FALSE)
rgl(n, sigma = 1, shape = 1)
```

Arguments

x	vector of quantiles.
p	vector of probabilities.
n	number of observations.
sigma	positive scale parameter.
shape	shape parameter.
log, log.p	logical; if TRUE, probabilities are log–transformed.
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ otherwise, $P[X > x]$.

Details

The univariate symmetric generalized Laplace distribution (Kotz et al, 2001, p.190) has density

$$f(x) = \frac{2}{\sqrt{2\pi}\Gamma(s)\sigma^{s+1/2}}(\frac{|x|}{\sqrt{2}})^{\omega}B_{\omega}(\frac{\sqrt{2}|x|}{\sigma})$$

where σ is the scale parameter, $\omega = s - 1/2$, and s is the shape parameter. Γ denotes the Gamma function and B_u the modified Bessel function of the third kind with index u. The variance is $s\sigma^2$.

This distribution is the univariate and symmetric case of MultivariateGenLaplace.

Value

dgl gives the density, pgl gives the distribution function, qgl gives the quantile function, and rgl generates random deviates.

Author(s)

Marco Geraci

References

Kotz, S., Kozubowski, T., and Podgorski, K. (2001). The Laplace distribution and generalizations. Boston, MA: Birkhauser.

See Also

MultivariateGenLaplace

Laplace *The Laplace Distribution*

Description

Density and random generation for the (symmetric) Laplace distribution.

Usage

```
dl(x, mu = 0, sigma = 1, log = FALSE)
rl(n, mu = 0, sigma = 1)
```

Arguments

x	vector of quantiles.
n	number of observations.
mu	location parameter.
sigma	positive scale parameter.
log	logical; if TRUE, probabilities are log–transformed.

Details

The Laplace distribution has density

$$f(x) = \frac{1}{\sqrt{2}\sigma} e^{-\frac{sqrt(2)}{\sigma}|x-\mu|}$$

where μ is the location parameter and σ is the scale parameter.

Value

dl gives the density and rl generates random deviates.

Author(s)

Marco Geraci

References

Kotz, S., Kozubowski, T., and Podgorski, K. (2001). The Laplace distribution and generalizations. Boston, MA: Birkhauser.

See Also

MultivariateLaplace, GenLaplace

---

logLik.nlmm *Extract Log-Likelihood*

Description

logLik.nlmm extracts the log-likelihood of a fitted nlmm.

Usage

```
## S3 method for class 'nlmm'
logLik(object, ...)
```

Arguments

object	an object of class "nlmm".
...	not used.

Value

Returns the loglikelihood of the fitted model. This is a number with at one attribute, "df" (degrees of freedom), giving the number of (estimated) parameters in the model.

Author(s)

Marco Geraci

See Also

nlmm

lrt_nlmm *Likelihood Ratio Test for Generalized Laplace Mixed-Effects Models*

Description

This function is used to perform a likelihood ratio test for two fitted generalized Laplace mixed-effects models.

Usage

```
lrt_nlmm(object0, object1)
## S3 method for class 'lrt_nlmm'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

object0	object of class nlmm with estimates of the constrained model.
object1	object of class nlmm with estimates of the unconstrained model.
x	a lrt_nlmm object.
digits	a non-null value for digits specifies the minimum number of significant digits to be printed in values.
...	not used.

Value

An object of class lrt_nlmm is a list containing the following components:

statistic	the value of the test statistic
p.value	the p-value of the test
df	either the degrees of freedom of a chi-squared test or the weights of a chi-bar-squared test
V	the matrix based on which the weights for the chi-bar-squared test are calculated
alpha	values of the shape parameter in the constrained object
alpha.index	index of the constrained shape parameter
chibar	logical flag. If TRUE, the test statistic is a chi-bar

Note

The function lrt_nlmm is a wrapper for routines developed by Alessio Farcomeni.

Author(s)

Marco Geraci and Alessio Farcomeni

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

See Also

nlmm

`MultivariateGenLaplace`

The Multivariate Asymmetric Generalized Laplace Distribution

Description

Density and random generation for the multivariate asymmetric generalized Laplace distribution.

Usage

```
dmgl(x, mu = rep(0, d), sigma = diag(d), shape = 1, log = FALSE)
rmgl(n, mu, sigma, shape = 1)
```

Arguments

`x`	vector of quantiles.
`n`	number of observations.
`mu`	simmetry parameter.
`sigma`	scale parameter – positive-definite matrix.
`shape`	shape parameter.
`log`	logical; if `TRUE`, probabilities are log–transformed.

Details

This is the distribution described by Kozubowski et al (2013) and has density

$$f(x) = \frac{2\exp(\mu'\Sigma^{-1}x)}{(2\pi)^{d/2}\Gamma(s)|\Sigma|^{1/2}}(\frac{Q(x)}{C(\Sigma,\mu)})^{\omega}B_{\omega}(Q(x)C(\Sigma,\mu))$$

where μ is the symmetry parameter, Σ is the scale parameter, $Q(x) = \sqrt{x'\Sigma^{-1}x}$, $C(\Sigma,\mu) = \sqrt{2+\mu'\Sigma^{-1}\mu}$, $\omega = s - d/2$, d is the dimension of x, and s is the shape parameter (note that the parameterization in nlmm is $\alpha = \frac{1}{s}$). Γ denotes the Gamma function and B_u the modified Bessel function of the third kind with index u. The parameter μ is related to the skewness of the distribution (symmetric if $\mu = 0$). The variance-covariance matrix is $s(\Sigma + \mu\mu')$. The multivariate asymmetric Laplace is obtained when $s = 1$ (see MultivariateLaplace).

In the symmetric case ($\mu = 0$), the multivariate GL distribution has two special cases: multivariate normal for $s \to \infty$ and multivariate symmetric Laplace for $s = 1$.

The **univariate symmetric** GL distribution is provided via GenLaplace, which gives the distribution and quantile functions in addition to the density and random generation functions.

Value

`dmgl` gives the GL density of a d-dimensional vector `x`. `rmgl` generates a sample of size `n` of d-dimensional random GL variables.

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

Kozubowski, T. J., K. Podgorski, and I. Rychlik (2013). Multivariate generalized Laplace distribution and related random fields. Journal of Multivariate Analysis 113, 59-72.

See Also

`GenLaplace`

`MultivariateLaplace` *The Multivariate Asymmetric Laplace Distribution*

Description

Density and random generation for the multivariate asymmetric Laplace distribution.

Usage

```
dmal(x, mu = rep(0, d), sigma = diag(d), log = FALSE)
rmal(n, mu, sigma)
```

Arguments

`x`	vector of quantiles.
`n`	number of observations.
`mu`	asymmetry parameter.
`sigma`	scale parameter – positive-definite matrix.
`log`	logical; if `TRUE`, probabilities are log–transformed.

Details

This is the multivariate extension of the (univariate) asymmetric Laplace distribution. It is a special case of `MultivariateGenLaplace` with `shape = 1`.

Author(s)

Marco Geraci

References

Kotz, S., Kozubowski, T., and Podgorski, K. (2001). The Laplace distribution and generalizations. Boston, MA: Birkhauser.

See Also

Laplace, MultivariateGenLaplace

---

nlmm *Fitting Generalized Laplace Mixed-Effects Models*

Description

nlmm is used to fit mixed-effects models based on the generalized Laplace distribution.

Usage

```
nlmm(fixed, random, group, covariance = "pdDiag", data = sys.frame(sys.parent()),
subset, weights = NULL, na.action = na.fail, control = list(), contrasts = NULL,
fit = TRUE)
```

Arguments

fixed	an object of class formula for fixed effects: a symbolic description of the model to be fitted.
random	a one-sided formula of the form ~x1 + x2 + ... + xn for random effects: a symbolic description of the model to be fitted.
group	grouping factor.
covariance	variance–covariance matrix of the random effects. Default is pdDiag (see details)
data	an optional data frame containing the variables named in fixed, random, group, and weights. By default the variables are taken from the environment from which nlmm is called.
subset	an optional vector specifying a subset of observations to be used in the fitting process.
weights	an optional varFunc object or one-sided formula describing the within-group heteroscedasticity structure. If given as a formula, it is used as the argument to varFixed, corresponding to fixed variance weights. See the documentation on varClasses in nlme package for a description of the available varFunc classes. Defaults to NULL, corresponding to homoscedastic within-group errors.
na.action	a function that indicates what should happen when the data contain NAs. The default action (na.fail) causes nlmm to print an error message and terminate if there are any incomplete observations.
control	list of control parameters of the fitting process. See nlmmControl.
contrasts	not yet implemented.
fit	logical flag. If FALSE the function returns a list of objects used for fitting.

Details

The function fits a generalized Laplace mixed-effects model conditional on the covariates, as specified by the `formula` argument, and on random effects, as specified by the `random` argument. The predictor is assumed to be linear. The function maximizes the (log)likelihood of the generalized Laplace regression as proposed by Geraci and Farcomeni (2020). The likelihood is numerically integrated via Gaussian quadrature techniques. The optimization algorithm can be either `optim` (Nelder-Mead by default) or `nlminb`. See `nlmmControl` for more details.

By default, the function fits a mixed-effects model where both random effects and error term follow a generalized Laplace distribution (`GenLaplace`). This is a family of distributions that includes the normal and the Laplace distributions as special cases. Constrained fitting can be controlled via the arguments `alpha.index` and `alpha` in `nlmmControl`. For example, if `alpha.index = 0`, the model is Normal-Normal if `alpha = c(0,0)`, Normal-Laplace if `alpha = c(0,1)`, Laplace-Normal if `alpha = c(1,0)`, and Laplace-Laplace if `alpha = c(1,1)`. But any value of alpha between 0 (normal distribution) and 1 (Laplace distribution) is allowed.

Different standard types of positive–definite matrices for the random effects can be specified: `pdIdent` multiple of an identity; `pdCompSymm` compound symmetry structure (constant diagonal and constant off–diagonal elements); `pdDiag` diagonal; `pdSymm` general positive–definite matrix, with no additional structure.

Within-group heteroscedasticity can be modeled via the `weights` argument using `varClasses` in the `nlme` packages.

Value

`nlmm` returns an object of `class` `nlmm`.

The function `summary` is used to obtain and print a summary of the results.

An object of class `nlmm` is a list containing the following components:

`theta`	a vector containing (in this order) fixed regression coefficients, parameters of the variance–covariance matrix of the random effects, shape parameter, scale parameter, and (optional) within-group variance function parameters. All parameters are unconstrained. See `VarCorr.nlmm` to extract the variance–covariance of the random effects from an "nlmm" object.
`theta_x`, `theta_z`	partition of `theta`: fixed regression coefficients (`theta_x`) and unique variance–covariance unconstrained parameters (`theta_z`).
`tau`	unconstrained shape parameter.
`alpha`	constrained shape parameter.
`phi`	unconstrained scale parameter.
`sigma`	constrained scale parameter.
`vf`	(fitted) within-group variance function of class `varFunc`. S3 methods (`summary.varFunc`, `varWeights.varFunc`, `coef.varFunc`) can be applied.
`value`	negative log–likelihood.
`call`	the matched call.
`nn`	column names of `mmf`.

mm	column names of mmr.
nobs	the number of observations.
dim_theta	the number of columns in mmf and mmr.
dim_theta_z	the length of theta_z.
mmf	the model matrix – fixed effects.
mmr	the model matrix – random effects.
y	the model response.
revOrder	original order of observations (now ordered according to group).
group	the grouping factor.
ngroups	the number of groups.
InitialPar	starting values for theta, included the fitted lme or lm object from where starting values have been taken.
control	list of control parameters used for optimization (see nlmmControl).
cov_name	class of variance-covariance matrix for the random effects.
mfArgs	arguments for model.frame to return the full data frame.
sc	model's distribution. "Generalized Laplace" if unconstrained estimation, or one of four special case for specific values of alpha ("Normal-Normal", "Normal-Laplace", "Laplace-Normal", "Laplace-Laplace").

Author(s)

Marco Geraci

References

Geraci M (2017). Mixed-effects models using the normal and the Laplace distributions: A 2 x 2 convolution scheme for applied research. arXiv:1712.07216v1 [stat.ME]. URL: https://arxiv.org/abs/1712.07216v1.

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

See Also

summary.nlmm, fixef.nlmm, ranef.nlmm, VarCorr.nlmm, predict.nlmm, residuals.nlmm, nlmmControl

Examples

```
data(rats)

nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats,
control = nlmmControl(multistart = FALSE))
```

nlmm.fit *Fitter Function for Generalized Laplace Mixed-Effects Models*

Description

This is the basic computing engine called by nlmm used to fit generalized Laplace mixed-effects models. These should usually not be used directly unless by experienced users.

Usage

```
nlmm.fit(args, control)
```

Arguments

args	list of arguments for fitting as returned by nlmm with fit = FALSE.
control	list of control parameters of the fitting process as returned by nlmmControl.

Details

The function fits a generalized Laplace mixed-effects model conditional on the covariates, as specified by the formula argument, and on random effects, as specified by the random argument. The predictor is assumed to be linear. The function maximizes the (log)likelihood of the generalized Laplace regression as proposed by Geraci and Farcomeni (2020). The likelihood is numerically integrated via Gaussian quadrature techniques. The optimization algorithm is based on the function nlminb.

By default, the function fits a mixed-effects model where both random effects and error term follow a generalized Laplace distribution (GenLaplace). This is a family of distributions that includes the normal and the Laplace distributions as special cases. Constrained fitting can be controlled via the arguments alpha.index and alpha in nlmmControl. For example, if alpha.index = 0, the model is Normal-Normal if alpha = c(0,0), Normal-Laplace if alpha = c(0,1), Laplace-Normal if alpha = c(1,0), and Laplace-Laplace if alpha = c(1,1). But any value of alpha between 0 (normal distribution) and 1 (Laplace distribution) is allowed.

Different standard types of positive–definite matrices for the random effects can be specified: pdIdent multiple of an identity; pdCompSymm compound symmetry structure (constant diagonal and constant off–diagonal elements); pdDiag diagonal; pdSymm general positive–definite matrix, with no additional structure.

Within-group heteroscedasticity can be modeled via the weights argument using varClasses in the nlme packages.

Value

nlmm returns an object of class nlmm.

The function summary is used to obtain and print a summary of the results.

An object of class nlmm is a list containing the following components:

theta	a vector containing (in this order) fixed regression coefficients, parameters of the variance–covariance matrix of the random effects, shape parameter, scale parameter, and (optional) within-group variance function parameters. All parameters are unconstrained. See VarCorr.nlmm to extract the variance–covariance of the random effects from an "nlmm" object.
theta_x, theta_z	partition of theta: fixed regression coefficients (theta_x) and unique variance–covariance unconstrained parameters (theta_z).
tau	unconstrained shape parameter.
alpha	constrained shape parameter.
phi	unconstrained scale parameter.
sigma	constrained scale parameter.
vf	(fitted) within-group variance function of class varFunc. S3 methods (summary.varFunc, varWeights.varFunc, coef.varFunc) can be applied.
value	negative log–likelihood.
call	the matched call.
nn	column names of mmf.
mm	column names of mmr.
nobs	the number of observations.
dim_theta	the number of columns in mmf and mmr.
dim_theta_z	the length of theta_z.
mmf	the model matrix – fixed effects.
mmr	the model matrix – random effects.
y	the model response.
revOrder	original order of observations (now ordered according to group).
group	the grouping factor.
ngroups	the number of groups.
InitialPar	starting values for theta, included the fitted lme or lm object from where starting values have been taken.
control	list of control parameters used for optimization (see nlmmControl).
cov_name	class of variance-covariance matrix for the random effects.
mfArgs	arguments for model.frame to return the full data frame.
sc	model's distribution. "Generalized Laplace" if unconstrained estimation, or one of four special case for specific values of alpha ("Normal-Normal", "Normal-Laplace", "Laplace-Normal", "Laplace-Laplace").

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research.

See Also

summary.nlmm, fixef.nlmm, ranef.nlmm, VarCorr.nlmm, predict.nlmm, residuals.nlmm

Examples

```
data(rats)

nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats, control = nlmmControl(multistart = FALSE))
```

---

nlmmControl *Control parameters for nlmm estimation*

Description

A list of parameters for controlling the fitting process.

Usage

```
nlmmControl(method = "Nelder-Mead", nK = 8, multistart = TRUE,
 grid = c(0.001, 0.5, 0.999), alpha = c(0.5, 0.5), alpha.index = 9,
 lme = TRUE, lmeMethod = "REML", lmeOpt = "nlminb", verbose = FALSE)
```

Arguments

method	character vector that specifies the optimization algorithm to fit a generalized Laplace mixed-effects model. The default is "Nelder-Mead".
nK	number of knots for each of the two quadratures.
multistart	logical flag. If TRUE (default), the algorithm is run with multiple starting values for the parameter alpha. See also grid.
grid	a vector of values for multi-start optimization. It can be used in conjunction with constrained estimation.
alpha	vector of length 2 with starting values between 0 and 1 for the parameter alpha (ignored if multistart is TRUE) or values at which alpha is constrained if alpha.index is one of 0, 1, or 2. The first element is for the shape parameter of the random effects, the second for the error term. See Geraci and Farcomeni (2020).

alpha.index	the estimation with parameter alpha is unconstrained if alpha.index is equal to 9 (default). If equal to 0, both shape parameters (random effects and error term) are constrained during estimation and set equal to alpha. If equal to 1, the first shape parameter (random effects) is constrained during estimation and set equal to the first element of alpha. If equal to 2, the second shape parameter (error term) is constrained during estimation and set equal to the second element of alpha.
lme	logical flag. Should lme be used to get starting values? If FALSE, lm is used instead.
lmeMethod	fitting method for lme when obtaining starting values. If "REML" the model is fit by maximizing the restricted log-likelihood. If "ML" the log-likelihood is maximized.
lmeOpt	optimization algorithm for lme. Either either nlminb (the default) or optim. This is passed to argument opt in lmeControl.
verbose	logical flag. If TRUE, information about the fitting process is printed out.

Details

The estimation algorithm for fitting generalized Laplace mixed-effects (GLME) models is described in Geraci and Farcomeni (2020). For unconstrained estimation, it is recommended to leave the default arguments in nlmmControl unchanged.

The integrated log-likelihood is maximized with either optim, in which case method has to be one of optim's options ("Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Brent"), or nlminb, in which case one must use method = "nlminb".

Since the parameter alpha is bidimensional, care should be taken when increasing the number of quadrature knots nK since the total number of quadrature points is given by 2^{nK}. For the same reason, care should be taken when providing the grid values for multi-start optimization since the total number of starting points will be s^2, where s = length(grid).

If alpha.index is 1 (or 2), the first (or second) element of the alpha parameter is constrained during estimation and set equal to the corresponding value of alpha. The element of the alpha parameter that is *unconstrained* is initialized with the corresponding element of alpha (if multistart is FALSE) or with values in grid (if multistart is TRUE).

If alpha.index is 0, both elements of the alpha parameter are fixed and set equal to alpha. In this case, the argument multistart is ignored. If alpha is c(0,0), the corresponding model is Normal-Normal and lme is used for fitting (only via maximum likelihood). Note that in this case, lmeOpt can still be used.

Value

a list of control parameters.

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

See Also

nlmm

---

predict.nlmm *Predictions from an* nlmm *Object*

Description

The predictions at level 0 correspond to predictions based only on the fixed effects estimates. The predictions at level 1 are obtained by adding the best linear predictions of the random effects to the predictions at level 0.

Usage

```
## S3 method for class 'nlmm'
predict(object, level = 0, ...)
```

Arguments

object	an nlmm object.
level	an integer vector giving the level of grouping to be used in obtaining the predictions.
...	not used.

Value

a vector of predictions.

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

See Also

nlmm, ranef.nlmm, fixef.nlmm

Examples

```
## Not run:
data(rats)
fit <- nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats,
control = nlmmControl(multistart = FALSE))

# Individual growth trajectories
predict(fit, level = 1)

## End(Not run)
```

`print.nlmm` *Print an* `nlmm` *Object*

Description

Print an object generated by `nlmm`.

Usage

```
## S3 method for class 'nlmm'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`x`	an `nlmm` object.
`digits`	a non-null value for digits specifies the minimum number of significant digits to be printed in values.
`...`	not used.

Author(s)

Marco Geraci

See Also

`nlmm`

print.summary.nlmm *Print an* nlmm *Summary Object*

Description

Print summary of an nlmm object.

Usage

```
## S3 method for class 'summary.nlmm'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

x	a summary.nlmm object.
digits	a non-null value for digits specifies the minimum number of significant digits to be printed in values.
...	not used.

Author(s)

Marco Geraci

See Also

nlmm, summary.nlmm

ranef.nlmm *Extract Random Effects*

Description

This function computes random effects for a linear quantile mixed model.

Usage

```
## S3 method for class 'nlmm'
ranef(object, ...)
```

Arguments

object	an object of class nlmm.
...	not used.

Details

The prediction of the random effects is done via estimated best linear prediction (Geraci and Farcomeni, 2019). The generic function `ranef` is imported from the `nlme` package (Pinheiro et al, 2014).

Value

a data frame of predicted random effects.

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

Pinheiro J, Bates D, DebRoy S, Sarkar D and R Core Team (2014). nlme: Linear and Nonlinear Mixed Effects Models. R package version 3.1-117, https://CRAN.R-project.org/package=nlme.

See Also

`nlmm`, `fixef.nlmm`

Examples

```
## Not run:
data(rats)
fit <- nlmm(y ~ trt*time, random = ~ time, group = id, data = rats,
control = nlmmControl(multistart = FALSE))

# Predicted random intercepts and slopes
ranef(fit)

## End(Not run)
```

---

rats *Growth curves*

Description

The `rats` data frame has 135 rows and 4 columns of the change in weight measured over time for rats assigned to different treatment groups.

Format

This data frame contains the following columns:

id grouping variable.

time time (week) of measurement (0, 1, 2, 3, 4).

trt treatment group (1, 2, 3).

y weight (grams)

Details

In a weight gain experiment, 30 rats were randomly assigned to three treatment groups: treatment 1, a control (no additive); treatments 2 and 3, which consisted of two different additives (thiouracil and thyroxin respectively) to the rats drinking water (Box, 1950). Weight (grams) of the rats was measured at baseline (week 0) and at weeks 1, 2, 3, and 4. Data on three of the 10 rats from the thyroxin group were subsequently removed due to an accident at the beginning of the study.

Source

G. E. P. Box, Problems in the analysis of growth and wear curves, Biometrics 6 (4) (1950) 362-389.

`residuals.nlmm` *Residuals from an* `nlmm` *Object*

Description

The residuals at level 0 correspond to population residuals (based only on the fixed effects estimates). The residuals at level 1 are obtained by adding the best linear predictions of the random effects to the predictions at level 0 and the subtracting these from the model response.

Usage

```
## S3 method for class 'nlmm'
residuals(object, level = 0, ...)
```

Arguments

`object`	an `nlmm` object.
`level`	an optional integer vector giving the level of grouping to be used in obtaining the predictions. Level zero corresponds to the population residuals.
`...`	not used.

Value

a matrix of residuals.

Author(s)

Marco Geraci

References

Geraci, M. and Farcomeni A. (2020). A family of linear mixed-effects models using the generalized Laplace distribution. Statistical Methods in Medical Research, 29(9), 2665-2682.

See Also

nlmm, predict.nlmm, fixef.nlmm, ranef.nlmm,

---

summary.nlmm *Summary for an* nlmm *Object*

Description

Summary method for class nlmm.

Usage

```
## S3 method for class 'nlmm'
summary(object, alpha = 0.05, ...)
```

Arguments

object	an object of class nlmm.
alpha	significance level.
...	not used.

Details

print.summary.nlmm formats the coefficients, standard errors, etc. and additionally gives 'significance stars'.

Value

an object of class summary.nlmm. The function summary.nlmm computes and returns a list of summary statistics of the fitted generalized Laplace mixed-effects model given in object, using the components (list elements) from its argument, plus

tTable	a matrix with estimates, standard errors, etc.

Author(s)

Marco Geraci

See Also

print.summary.nlmm nlmm

Examples

```
## Not run:
data(rats)
fit <- nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats,
control = nlmmControl(multistart = FALSE))
summary(fit)


## End(Not run)
```

VarCorr.nlmm *Extract Variance-Covariance Matrix*

Description

This function extracts the variance-covariance matrix of the random effects from a fitted nlmm object.

Usage

```
## S3 method for class 'nlmm'
VarCorr(x, sigma = NULL, ...)
```

Arguments

x	an object of class "nlmm".
sigma	not used.
...	not used.

Details

This function returns the variance or the variance-covariance matrix of the random effects. The generic function VarCorr is imported from the nlme package (Pinheiro et al, 2014).

Author(s)

Marco Geraci

References

Pinheiro J, Bates D, DebRoy S, Sarkar D and R Core Team (2014). nlme: Linear and Nonlinear Mixed Effects Models. R package version 3.1-117, https://CRAN.R-project.org/package=nlme.

See Also

nlmm

Examples

```
## Not run:
data(rats)
fit <- nlmm(y ~ trt*time, random = ~ time, group = id, data = rats, cov = "pdSymm",
control = nlmmControl(multistart = FALSE))

# Symmetric variance-covariance of random intercepts and slopes
VarCorr(fit)

## End(Not run)
```

---

vcov.nlmm *Calculate Variance-Covariance Matrix for a Fitted Generalized Laplace Mixed-Effects Object*

Description

Returns the variance-covariance matrix of the all the parameters of a fitted nlmm object.

Usage

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

Arguments

object	an nlmm object.
...	not used.

Details

Gives the variance-covariance matrix of the GLME estimator, on the scale of the unconstrained, unrestricted parameters. The size is dxd, $d = p + r + 2 + 1 + s$, with p fixed coefficients, r non-redundant parameters of the random effects distribution, 2 shape parameters, 1 scale parameter, s parameters of the residual variance function (if specified in the model), in this order.

Value

a matrix.

Author(s)

Marco Geraci

See Also

nlmm

Examples

```
## Not run:
data(rats)

# Number of parameters is d = 6 + 3 + 2 + 1 + 0 = 12
fit <- nlmm(y ~ trt*time, random = ~ time, group = id, data = rats,
cov = "pdSymm", control = nlmmControl(multistart = FALSE))
fit$par
vcov(fit)

# Number of parameters is d = 6 + 1 + 2 + 1 + 4 = 14
fit <- nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats,
control = nlmmControl(multistart = FALSE), weights = varIdent(form = ~ 1|time))
fit$par
vcov(fit)

# Number of parameters is d = 6 + 1 + 0 + 1 + 0 = 8
# Note that the shape parameters are now constrained
fit <- nlmm(y ~ trt*time, random = ~ 1, group = id, data = rats,
control = nlmmControl(alpha.index = 0, multistart = FALSE))
fit$par
vcov(fit)


## End(Not run)
```

Index