Package ‘lqmix’

May 8, 2026

Type Package

Title Linear Quantile Mixture Models

Description Estimate linear quantile mixtures based on Time-Constant (TC) and/or Time-Varying (TV), discrete, random coefficients.

Date 2025-10-16

Version 1.2

Author Maria Francesca Marino [aut, cre],
Marco Alfo' [aut],
Nicola Salvati [aut],
Maria Giovanna Ranalli [aut]

Maintainer Maria Francesca Marino <mariafrancesca.marino@unifi.it>

Depends R (>= 3.5.0)

License GPL (>= 2)

Encoding UTF-8

LazyData TRUE

Imports quantreg, stats, methods, Rdpack, diagram, doParallel, foreach, doSNOW, doRNG

Suggests testthat (>= 3.0.0)

RdMacros Rdpack

RoxygenNote 7.3.3

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2025-10-17 09:20:02 UTC

Contents

lqmix-package 2
cd4 4
coef.lqmix 5

coef.lqr 5
coef.search_lqmix 6
dal 6
logLik.lqmix 7
logLik.lqr 8
logLik.search_lqmix 8
lqmix 9
lqr 12
pain 13
plot.lqmix 14
plot.search_lqmix 15
predict.lqmix 15
predict.lqr 16
predict.search_lqmix 16
print.lqmix 17
print.lqr 18
print.search_lqmix 18
print.summary.lqmix 19
print.summary.lqr 19
print.summary.search_lqmix 20
residuals.lqmix 20
residuals.lqr 21
residuals.search_lqmix 22
search_lqmix 22
summary.lqmix 24
summary.lqr 26
summary.search_lqmix 27
varAL 28
vcov.lqmix 29
vcov.lqr 29
vcov.search_lqmix 30

Index **31**

`lqmix-package` *Overview of the package* `lqmix`

Description

The `lqmix` package allows for the estimation of finite mixtures of linear quantile regression models based on Time-Constant (TC) and/or Time-Varying (TV), discrete, random coefficients for the analysis of longitudinal data.

Details

lqmix is an R package devoted to the estimation of a class of linear quantile regression models for longitudinal data, in the presence of Time-Constant (TC) and/or Time-Varying (TV), unit-specific, random coefficients, having unspecific distribution. The parameters of this distribution, together with all the others characterizing the model, are estimated in a maximum likelihood framework, via an extended Expectation-Maximization (EM) algorithm. This approach leads to the estimation of discrete distributions for the random coefficients, which give rise to a likelihood function similar to that of standard finite mixture models (in the case of TC random coefficients only), hidden Markov models (in the case of TV random coefficients only), or mixed hidden Markov models with discrete effects (in the case of both TC and TV random coefficients).

Parameters' standard errors are estimated via a block-bootstrap procedure, while model selection is performed by either maximizing the log-likelihood function, or minimizing the Akaike Information Criterion, or the Bayesian Information Criterion.

Missing data are allowed and treated under a Missing at Random assumption.

Author(s)

Maria Francesca Marino [aut,cre], Marco Alfo' [aut], Nicola Salvati [aut], and Maria Giovanna Ranalli [aut]

Maintainer: Maria Francesca Marino <mariafrancesca.marino@unifi.it>

References

Alfo' M, Salvati N, Ranalli MG (2017). "Finite Mixtures of Quantiles and M-quantile models." *Statistics and Computing*, **27**, 547-570.

Aitkin M (1996). "A general maximum likelihood analysis of overdispersion in generalized linear models." *Statistics and Computing*, **6**, 251-262.

Aitkin M (1999). "A general maximum likelihood analysis of variance components in generalized linear models." *Biometrics*, **55**, 117–128.

Farcomeni A (2012). "Quantile regression for longitudinal data based on latent Markov subject-specific parameters." *Statistics and Computing*, **22**.

Bartolucci F, Farcomeni A, Pennoni F (2012). *Latent Markov models for longitudinal data*. Taylor & Francis.

Zucchini W, MacDonald IL, Langrock R (2017). *Hidden Markov models for time series*, Monographs on Statistics and Applied Probability. Chapman and Hall/CRC.

Marino MF, Tzavidis N, Alfo' M (2018). "Mixed hidden Markov quantile regression models for longitudinal data with possibly incomplete sequences." *Statistical Methods in Medical Research*, **27**, 2231-2246.

Altman RJ (2007). "Mixed hidden Markov models: an extension of the hidden Markov model to the longitudinal data setting." *Journal of the American Statistical Association*, **102**, 201–210.

Maruotti A (2011). "Mixed Hidden Markov Models for Longitudinal Data: An Overview." *International Statistical Review*, **79**. ISSN 1751-5823.

cd4 *CD4 Data*

Description

The cd4 data frame is made by a total of 2376 rows and 8 columns providing information on CD4 cell counts of 369 subjects followed for a maximum of 12 measurement occasions.

Usage

```
data(cd4)
```

Format

A data frame with 2376 observations on the following 8 variables:

`sbj.id` subject id

`time.id` time id

`count` CD4 count

`lcount` log(CD4 count + 1)

`time` years since seroconversion

`age` age (yrs) centered around 30

`packs` packs of cigarettes per day

`partners` number of sexual partners

`drugs` recreational drug use indicator

`cesd` depression score

Details

Multi-center AIDS Cohort Study providing a total of 2376 CD4+ cell counts of 369 HIV-infected men covering a period of approximately eight and half years. The number of measurements for each subject varies from 1 to 12. The CD4+ cell data are highly unbalanced.

References

Zeger SL, Diggle PJ (1994). "Semiparametric models for longitudinal data with application to CD4 cell numbers in HIV seroconverters." *Biometrics*, **50**, 689–699.

coef.lqmix *Print the estimated coefficients of an* `lqmix` *object*

Description

Print the estimated coefficients of a fitted model stored in an object of `class` `lqmix`.

Usage

```
## S3 method for class 'lqmix'
coef(object, ...)
```

Arguments

`object`	an `lqmix` object
`...`	not used

Value

Return the estimated coefficients in the longitudinal data model obtained at convergence of the EM algorithm for a fitted model of `class` `lqmix`.

coef.lqr *Print the estimated coefficients of an* `lqr` *object*

Description

Print the estimated coefficients of a fitted model stored in an object of `class` `lqr`.

Usage

```
## S3 method for class 'lqr'
coef(object, ...)
```

Arguments

`object`	an `lqmix` object
`...`	not used

Value

Return the estimated coefficients for a fitted model of `class` `lqr`.

coef.search_lqmix — *Printn the estimated coefficients for the optimal fitted model*

Description

Print the estimated coefficients of the optimal fitted model stored in an object of class search_lqmix.

Usage

```
## S3 method for class 'search_lqmix'
coef(object, ...)
```

Arguments

object	a search_lqmix object
...	not used

Value

Return the estimated coefficients obtained at convergence of the EM algorithm for the optimal model obtained at convergence of the EM algorithm for a fitted model of class search_lqmix.

dal — *Density of the Asymmetric Laplace Distribution*

Description

Compute the density for the three parameter Asymmetric Laplace Distribution.

Usage

```
dal(y, mu = 0, sigma = 1, qtl = 0.5, log = FALSE)
```

Arguments

y	vector of quantiles
mu	location parameter
sigma	scale parameter
qtl	skewness parameter
log	logical; if TRUE, probabilities are log-transformed

Details

The function computes the density of the Asymmetric Laplace distribution, with location μ, scale $\sigma > 0$ and skewness `qtl = q` in (0,1), as discussed by Koenker and Machado (1999) and Yu and Moyeed (2001), according to the following expression

$$f(y|\mu, \sigma, q) = \frac{q(1-q)}{\sigma} \exp(-\rho_q(\frac{y-\mu}{\sigma}))$$

Value

Return the density for the Asymmetric Laplace Distribution.

References

Koenker R, Machado JAF (1999). "Goodness of fit and related inference processes for quantile regression." *Journal of the American Statistical Association*, **94**, 1296–1310.

Yu K, Moyeed RA (2001). "Bayesian quantile regression." *Statistics & Probability Letters*, **54**, 437–447.

Yu K, Zhang J (2005). "A three-parameter asymmetric Laplace distribution and its extension." *Communications in Statistics. Theory and Methods*, **34**, 1867–1879.

`logLik.lqmix` *Print the log-likelihood of an* `{lqmix}` *object*

Description

Print the log-likelihood of a fitted model of `class lqmix`.

Usage

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

Arguments

`object`	an `{lqmix}` object
`...`	not used

Value

Return an object of `class logLik` providing the log-likelihood value at convergence of the EM algorithm for a fitted model of `class lqmix`.

logLik.lqr *Print the log-likelihood of an* lqr *object*

Description

Print the log-likelihood of a fitted model of class lqr.

Usage

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

Arguments

object	an lqr object
...	not used

Value

Return an object of class logLik providing the log-likelihood for a fitted model of class lqr.

logLik.search_lqmix *Print the log-likelihood of the optimal model stored in a* search_lqmix *object*

Description

Print the log-likelihood of an optimal fitted model stored in an object of class search_lqmix.

Usage

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

Arguments

object	an lqmix object
...	not used

Value

Return an object of class logLik providing the log-likelihood value at convergence of the EM algorithm for a fitted model of class lqmix.

lqmix *Linear Quantile Mixture with TC and/or TV, discrete, random coefficients*

Description

Estimate a finite mixture of linear quantile regression models with TC and/or TV discrete random coefficients, for a given number of components and/or states.

Usage

```
lqmix(formula, randomTC = NULL, randomTV = NULL, group, time, G = NULL,
  m = NULL, data, qtl = 0.5, eps = 10^-5, maxit = 1000, se = TRUE,
  R = 200, start = 0, parInit = list(betaf = NULL, betarTC = NULL,
  betarTV = NULL, pg = NULL, delta = NULL, Gamma = NULL, scale = NULL),
  verbose = TRUE, seed = NULL, parallel = FALSE, ncores = 2)
```

Arguments

`formula`	an object of class `formula`: a symbolic description of the model to be fitted
`randomTC`	a one-sided formula of the form `~z1+z2+...+zd`, where `z1,...,zd` denote the variables associated to TC random coefficients (1 for the intercept)
`randomTV`	a one-sided formula of the form `~w1+w2+...+wl`, where `w1,...,wl` denote the variables associated to TV random coefficients (1 for the intercept). Note that only TC variables are allowed
`group`	a string indicating the grouping variable, i.e., the factor identifying the unit longitudinal measurements refer to
`time`	a string indicating the time variable
`G`	number of mixture components associated to TC random coefficients
`m`	number of states associated to TV random coefficients
`data`	a data frame containing the variables named in `formula`, `randomTC`, `randomTV`, `group`, and `time`
`qtl`	quantile to be estimated
`eps`	tolerance level for convergence of the EM algorithm
`maxit`	maximum number of iterations for the EM algorithm
`se`	if set to TRUE, standard error are computed
`R`	number of bootstrap samples for computing standard errors
`start`	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
`parInit`	list of initial model parameters when `start=2`
`verbose`	if set to FALSE, no printed output is given during the function execution
`seed`	an integer value for random numbers generation, used for random parameter initialization and bootstrap standard errors
`parallel`	if set to TRUE, a parallelized code is use for standard error computation (if `se = TRUE`)
`ncores`	number of cores used for computing bootstrap standard errors (if required)

Details

The function computes ML estimates for a linear quantile mixture model with TC and/or TV random coefficients. Estimates are derived by maximizing the (log-)likelihood of an Asymmetric Laplace regression, where the location parameter is modeled as a function of fixed coefficients, together with TC, TV, or TC and TV discrete random coefficients, as proposed by Alfo' et. al (2017), Farcomeni (2012), and Marino et. al (2018), respectively.

The function requires data in long-format and two additional columns indicating the group identifier and the time occasion. The model is specified by means of the arguments `formula`, `formulaTC`, and `formulaTV`: `formula` is associated to fixed coefficients; `formulaTC` is associated to TC random coefficients; `formulaTV` is associated to TV random coefficients. In this latter, only TC variables (predictors) are allowed.

The function allows for missing data, including dropouts (monotone missing data) and intermittent missingness, under a missing-at-random assumption. Note that, when TV random coefficients are considered, intermittent missingness may cause biased inference.

If `se=TRUE`, standard errors based on a block bootstrap procedure are computed.

Value

Return an object of `class lqmix`. This is a list containing the following elements:

`betaf`	a vector of fixed regression coefficients
`betarTC`	a matrix of TC random coefficients, if present in the model
`betarTV`	a matrix of TV random coefficients, if present in the model
`pg`	the prior probabilities of the finite mixture associated to TC random coefficients, if present in the model
`delta`	the initial probability vector of the hidden Markov chain associated to TV random coefficients, if present in the model
`Gamma`	the transition probability matrix of the hidden Markov chain associated to TV random coefficients, if present in the model
`scale`	the scale parameter
`sigma.e`	the standard deviation of error terms
`lk`	the log-likelihood at convergence of the EM algorithm
`npar`	the total number of model parameters
`aic`	the AIC value
`bic`	the BIC value
`qtl`	the estimated quantile
`G`	the number of mixture components associated to TC random coefficients (if present)
`m`	the number of hidden states associated to TV random coefficients (if present)
`nsbjs`	the number of subjects (units)
`nobs`	the total number of observations
`se.betaf`	the standard errors for fixed regression coefficients

se.betarTC	the standard errors for TC random coefficients, if present in the model
se.betarTV	the standard errors for TV random coefficients, if present in the model
se.Mprob	the standard errors for the prior probabilities of the finite mixture associated to TC random coefficients, if present in the model
se.Init	the standard errors for the initial probabilities of the hidden Markov chain associated to TV random coefficients, if present in the model
se.Trans	the standard errors for the transition probabilities of the hidden Markov chain associated to TV random coefficients, if present in the model
se.scale	the standard error for the scale parameter
vcov	the bootstrap variance-covariance matrices of the regression coefficients
postTC	a matrix of size `nsbjs` x `G` with the estimated posterior probabilities for the finite mixture components associated to TC random coefficients, if present in the model
postTV	a matrix of size `nObs` x `m` with the estimated posterior probabilities for the hidden states associated to TV random coefficients, if present in the model
miss	the missingness type
model	the estimated model
call	the matched call
mmf	the model matrix associated with fixed regression coefficients
mmrTC	the model matrix associated with TC random coefficients, if present in the model
mmrTV	the model matrix associated with TV random coefficients, if present in the model
y	the model response
formula	the fixed model formula
randomTC	the model formula for the TC random coefficients, if present in the model
randomTV	the model formula for the TV random coefficients, if present in the model
group	the grouping variable
time	the time variable

References

Alfo' M, Salvati N, Ranalli MG (2017). "Finite Mixtures of Quantiles and M-quantile models." *Statistics and Computing*, **27**, 547-570.

Farcomeni A (2012). "Quantile regression for longitudinal data based on latent Markov subject-specific parameters." *Statistics and Computing*, **22**.

Marino MF, Tzavidis N, Alfo' M (2018). "Mixed hidden Markov quantile regression models for longitudinal data with possibly incomplete sequences." *Statistical Methods in Medical Research*, **27**, 2231-2246.

Examples

```
outTC = lqmix(formula=meas~trt+time+trt:time,randomTC=~1,
              group="id",time="time",G=2,data=pain,se=TRUE,R=10)


outTV = lqmix(formula=meas~trt+time+trt:time,randomTV=~1,
              group="id",time="time",m=2,data=pain,R=10)

outTCTV = lqmix(formula=meas~trt+time+trt:time,randomTC=~time,
                randomTV=~1,group="id",time="time",m=2,G=2,data=pain,R=10)
```

---

lqr *Linear Quantile Regression*

Description

Estimate a linear quantile regression model for independent data (no random coefficients).

Usage

```
lqr(formula, data, qtl = 0.5, se = TRUE, R = 100, verbose = TRUE,
  seed = NULL, parallel = FALSE, ncores = 2, ...)
```

Arguments

`formula`	an object of class `formula`: a symbolic description of the model to be fitted
`data`	a data frame containing the variables named in `formula`
`qtl`	quantile to be estimated
`se`	standard error computation
`R`	number of bootstrap samples for computing standard errors
`verbose`	if set to FALSE, no printed output is given during the function execution
`seed`	an integer value for random numbers generation, used for bootstrap standard errors
`parallel`	if set to TRUE, a parallelized code is use for standard error computation (if se=TRUE)
`ncores`	number of cores used for computing bootstrap standard errors (if required)
`...`	not used

Details

The function computes ML estimates for the parameters of a linear quantile regression model for independent observations. Estimates are derived by maximizing the (log-)likelihood of a Laplace regression, where the location parameter is modeled as a function of fixed coefficients only.

If `se=TRUE`, standard errors based on a bootstrap procedure are computed.

Value

Return an object of class lqr. This is a list containing the following elements:

betaf	a vector containing fixed regression coefficients
scale	the scale parameter
sigma.e	the standard deviation of error terms
lk	the log-likelihood
npar	the total number of model parameters
AIC	the AIC value
BIC	the BIC value
qtl	the estimated quantile
nobs	the total number of observations
se.betaf	the standard errors for the regression coefficients
se.scale	the standard error for the scale parameter
model	the estimated model
mmf	the model matrix associated to the regression coefficients
y	the model response
call	the matched call
formula	the model formula

References

Koenker R, Bassett Jr G (1978). "Regression Quantiles." *Econometrica*, **46**, 33–50.

Examples

```
out0 = lqr(formula=meas~trt+time+trt:time,data=pain,se=TRUE,R=10)
```

---

pain *Pain Data*

Description

The pain data frame consists of a total of 357 rows and 4 columns providing information on pain levels of 83 women in labor, followed for up 6 measurement occasions.

Usage

```
data(pain)
```

Format

A data frame with 357 observations on the following 5 variables:

`id` woman id

`meas` a numeric vector of self-reported pain scores on a 100mm line

`trt` a dummy variable with values 1 for subjects who received a pain medication and 0 for subjects who received a placebo

`time` a numeric vector of times (minutes since randomization) at which pain was measured

Details

The data set consists of repeated measurements of self-reported pain on n = 83 women. 43 women were randomly assigned to a pain medication group and 40 to a placebo group. The response was measured every 30 minutes on a 100-mm line: 0 means no pain and 100 means extreme pain. The number of measurements for each woman varies from 1 to 6. Data are severely skewed, and the skewness changes magnitude, and even sign, over time.

References

Davis CS (1991). "Semi-parametric and non-parametric methods for the analysis of repeated measurements with applications to clinical trials." *Statistics in Medicine*, **10**, 1959–1980.

`plot.lqmix` *Plots for* `lqmix` *objects*

Description

Graphically display component and/or transition probabilities of a fitted model of `class` `lqmix`.

Usage

```
## S3 method for class 'lqmix'
plot(x, ...)
```

Arguments

`x`	an object of class `lqmix`
`...`	not used

`plot.search_lqmix` *Plots for* `search_lqmix` *objects*

Description

Graphically display model selection criteria and component and/or transition probabilities of the optimal fitted model of `class search_lqmix`.

Usage

```
## S3 method for class 'search_lqmix'
plot(x, ...)
```

Arguments

`x`	an object of class `search_lqmix`
`...`	not used

`predict.lqmix` *Predictions from an* `lqmix` *object*

Description

Returns the predicted values for an object of class `lqmix`.

Usage

```
## S3 method for class 'lqmix'
predict(object, newdata = NULL, ...)
```

Arguments

`object`	an `lqmix` object
`newdata`	an optional data frame in which to look for variables with which to predict. If omitted, the fitted values are produced
`...`	not used

Details

The function computes predictions for an object of `class lqmix`. If the fitted model is based on TC, discrete, random coefficients only, a matrix of size `nsbjs x G` is given as output; if the fitted model is based on TV, discrete, random coefficients only, a matrix of size `nObs x m`; if the fitted model is based on both TC and TV, discrete, random coefficients, an array of size `nObs x G x m` is returned.

Value

A matrix or an array of predictions, based on the estimated model.

predict.lqr *Predictions from an lqr object*

Description

Returns the predicted values for an object of class lqr.

Usage

```
## S3 method for class 'lqr'
predict(object, newdata = NULL, ...)
```

Arguments

object	an lqr object
newdata	an optional data frame in which to look for variables with which to predict. If omitted, the fitted values are produced
...	not used

Details

The function computes predictions for an object of class lqr.

Value

A vector of predictions.

predict.search_lqmix *Predictions from the optimal fitted model*

Description

Returns the predicted values of the optimal fitted model stored in an object of class search_lqmix.

Usage

```
## S3 method for class 'search_lqmix'
predict(object, newdata = NULL, ...)
```

Arguments

object	a search_lqmix object
newdata	an optional data frame in which to look for variables with which to predict. If omitted, the fitted values are produced
...	not used

Details

The function computes predictions for the optimal fitted model stored in an object of class `search_lqmix`. If the optimal fitted model is based on TC, discrete, random coefficients only, a matrix of size `nsbjs x G` is given as output; if the optimal fitted model is based on TV, discrete, random coefficients only, a matrix of size `nObs x m`; if the optimal fitted model is based on both TC and TV, discrete, random coefficients, an array of size `nObs x G x m` is returned.

Value

A vector, a matrix, or an array of predictions, based on the optimal fitted model.

`print.lqmix` *Print an* `lqmix` *object*

Description

Print an object of `class lqmix`.

Usage

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

Arguments

`x`	an `lqmix` object
`digits`	a non-null value for digits specifying the minimum number of significant digits to be printed
`...`	not used

Value

Return an `lqmix` object.

print.lqr *Print an lqr object*

Description

Print an object of class lqr.

Usage

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

Arguments

x	an lqr object
digits	a non-null value for digits specifying the minimum number of significant digits to be printed
...	not used

Value

Return an lqr object.

print.search_lqmix *Print a search_lqmix object*

Description

Print an object of class search_lqmix.

Usage

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

Arguments

x	a search_lqmix object
digits	a non-null value for digits specifying the minimum number of significant digits to be printed
...	not used

Value

Return a search_lqmix object.

print.summary.lqmix — *Print the summary of an* lqmix *object*

Description

Print the summary of an object of class lqmix.

Usage

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

Arguments

x	a summary of an lqmix object
digits	a non-null value for digits specifying the minimum number of significant digits to be printed
...	not used

Value

Return a summary of an lqmix object.

print.summary.lqr — *Print the summary of an* lqr *object*

Description

Print the summary of an an object of class lqr.

Usage

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

Arguments

x	a summary of an lqr object
digits	a non-null value for digits specifying the minimum number of significant digits to be printed
...	not used

Value

Return a summary of an lqr object.

print.summary.search_lqmix
Print the summary of a search_lqmix *object*

Description

Print the summary of an object of class search_lqmix.

Usage

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

Arguments

x	a summary of a search_lqmix object
digits	a non-null value for digits specifying the minimum number of significant digits to be printed
...	not used

Value

Return a summary of a search_lqmix object.

residuals.lqmix *Residuals from an* lqmix *object*

Description

Returns the residuals from a fitted lqmix object.

Usage

```
## S3 method for class 'lqmix'
residuals(object, ...)
```

Arguments

object	an lqmix object
...	not used

Details

The function computes residuals for an object of class `lqmix`. If the fitted model is based on TC, discrete, random coefficients only, a matrix of size `nsbjs` x `G` is given as output; if the fitted model is based on TV, discrete, random coefficients only, a matrix a matrix of size `nObs` x `m` is given as output is given as output; if the fitted model is based on both TC and TV, discrete, random coefficients, an array of size `nObs` x `G` x `m` is returned. If the estimated model is based on TC, discrete, random coefficients only, a matrix is given as output. The number of columns corresponds to the estimated number of components (G). If the estimated model is based on TV, discrete, random coefficients only, a matrix is given as output. The number of columns corresponds to the estimated number of states (m). If the estimated model is based on both TC and TV, discrete, random coefficients, an array is given as output. The second and third dimensions correspond to the estimated number of components (G) and states (m), respectively.

Value

A matrix or an array of of residuals, based on the estimated model.

`residuals.lqr` *Residuals from an* `lqr` *object*

Description

Returns the residuals from a fitted `lqr` object.

Usage

```
## S3 method for class 'lqr'
residuals(object, ...)
```

Arguments

`object`	an `lqr` object
`...`	not used

Details

The function computes residuals for an object of `class` `lqr`.

Value

A vector of residuals.

`residuals.search_lqmix` *Residuals from the optimal fitted model*

Description

Returns the residuals from the optimal fitted model stored in an object of `class` `search_lqmix`.

Usage

```
## S3 method for class 'search_lqmix'
residuals(object, ...)
```

Arguments

`object`	an `search_lqmix` object
`...`	not used

Details

The function computes residuals for the optimal fitted model stored in an object of `class` `search_lqmix`. If the optimal fitted model is based on TC, discrete, random coefficients only, a matrix of size `nsbjs x G` is given as output; if the optimal fitted model is based on TV, discrete, random coefficients only, a matrix a matrix of size `nObs x m` is given as output is given as output; if the optimal fitted model is based on both TC and TV, discrete, random coefficients, an array of size `nObs x G x m` is returned.

Value

A vector, a matrix, or an array of of residuals, based on the estimated model.

`search_lqmix` *Search the Global Maximum of a Linear Quantile Mixture*

Description

Search the global maximum of the log-likelihood function for a finite mixture of linear quantile regression models with TC and/or TV, discrete, random coefficients, for varying number of components and/or states.

Usage

```
search_lqmix(formula, randomTC = NULL, randomTV = NULL, group, time,
  Gv = NULL, mv = NULL, data, method = "bic", nran = 0, qtl = 0.5,
  eps = 10^-5, maxit = 1000, se = TRUE, R = 200, verbose = TRUE,
  seed = NULL, parallel = FALSE, ncores = 2)
```

Arguments

formula	an object of class formula: a symbolic description of the model to be fitted
randomTC	a one-sided formula of the form ~z1+z2+...+zd, where z1,..., zd denote the variables associated to TC random coefficients (1 for the intercept)
randomTV	a one-sided formula of the form ~w1+w2+...+wl, where w1,..., wl denote the variables associated to TV random coefficients (1 for the intercept). Note that only TC variables are allowed
group	a string indicating the grouping variable, i.e., the factor identifying the unit longitudinal measurements refer to
time	a string indicating the time variable
Gv	vector of possible number of mixture components associated to TC random coefficients, if present in the model
mv	vector of possible number of states associated to the TV random coefficients, if present in the model
data	a data frame containing the variables named in formula, randomTC, randomTV, group, and time
method	method to use for selecting the optimal model. Possible values are "lk", "aic", or "bic"
nran	number of repetitions of each random initialization
qtl	quantile to be estimated
eps	tolerance level for convergence of the EM algorithm
maxit	maximum number of iterations for the EM algorithm
se	standard error computation for the optimal model
R	number of bootstrap samples for computing standard errors
verbose	if set to FALSE, no printed output is given during the function execution
seed	an integer value for random numbers generation, used for random parameter initialization and bootstrap standard errors
parallel	if set to TRUE, a parallelized code is use for standard error computation (if se = TRUE)
ncores	number of cores used for computing bootstrap standard errors (if required)

Details

The function allows to identify the optimal model specification in terms of number of mixture components and/or hidden states associated to TC and/or TV random coefficients, respectively. This is done by considering a multi-start strategy based on both deterministic and random starting points. The number or random starts is proportional to the number of mixture components and/or hidden states associated to the random coefficients in the model. For models based on TC random coefficients, nran x (G-1) random starts are considered; for models based on TV or TC and TV random coefficients, the number of random start is set to nran x (m-1) and nran x (G-1) x (m-1), respectively.

If `method = "lk"`, the optimal model selected by the function is that providing the highest log-likelihood value; if `method = "AIC"`, (`method = "BIC"`, respectively), the optimal model selected by the function is that providing the lowest AIC (BIC, respectively) value.

If `se=TRUE`, standard errors based on a block bootstrap procedure are computed for the identified optimal model.

Value

Return an object of `class search_lqmix`. This is a list containing the following elements:

`optimal`	the identified optimal model
`allmodels`	the output of each estimated model in the form of a list of lists; the former and the latter list are associated with TC and TV random coefficients, respectively
`lkv`	the vector of likelihood values for each fitted model
`aicv`	the vector of AIC values for each fitted model
`bicv`	the vector of BIC values for each fitted model
`qtl`	the estimated quantile
`mv`	the vector of possible number of states associated to TV random coefficients (if present)
`Gv`	the vector of possible number of mixture components associated to TC random coefficients (if present)
`method`	the method used to select the optimal model
`call`	the matched call

Examples

```
sTC = search_lqmix(formula=meas~trt+time+trt:time,
             randomTC=~1,group="id",time="time",Gv=1:3,method="bic",data=pain,se=FALSE)

sTV = search_lqmix(formula=meas~trt+time+trt:time,
randomTV=~1,group="id",time="time",mv=1:3,method="bic",data=pain,se=FALSE)

sTCTV = search_lqmix(formula=meas~trt+time+trt:time,
randomTC=~time,randomTV=~1,group="id",time="time",mv=1:3,Gv=1:3,method="bic",data=pain,se=FALSE)
```

`summary.lqmix` *Summary of an* `lqmix` *object*

Description

Summary method for the `class lqmix`.

Usage

```
## S3 method for class 'lqmix'
summary(object, ...)
```

Arguments

object	an lqmix object
...	not used

Value

Return an object of class summary.lqmix. This is a list of summary statistics for the fitted linear quantile mixture model given in object, with the following elements:

fix	a matrix with estimates, standard errors, Z statistics, and p-values for the fixed regression coefficients
ranTC	a matrix with estimates, standard errors, Z statistics, and p-values for the TC random coefficients (if present)
ranTV	a matrix with estimates, standard errors, Z statistics, and p-values for the TV random coefficients (if present)
pg	a matrix with estimates and standard errors for the prior probabilities of the finite mixture associated to TC random coefficients (if present)
delta	a matrix with estimates and standard errors for the initial probabilities of the hidden Markov chain associated to TV random coefficients (if present)
Gamma	a matrix with estimates and standard errors for the transition probabilities of the hidden Markov chain associated to TV random coefficients (if present)
scale	the scale parameter
sigma.e	the standard deviation of error terms
logLik	the log-likelihood at convergence of the EM algorithm
npar	the total number of model parameters
AIC	the AIC value
BIC	the BIC value
qtl	the estimated quantile
G	the number of mixture components associated to TC random coefficients, if present in the model
m	the number of hidden states associated to TV random coefficients, if present in the model
nsbj	the number of subjects
nobs	the total number of observations
miss	the missingness type
model	the estimated model
call	the matched call

summary.lqr *Summary of an* lqr *object*

Description

Summary method for the class lqr.

Usage

```
## S3 method for class 'lqr'
summary(object, ...)
```

Arguments

object	an lqr object
...	not used

Value

Return an object of class summary.lqr. This is a list of summary statistics for the fitted linear quantile regression model given in object, with the following elements:

fix	a matrix with estimates, standard errors, Z statistics, and p-values for the regression coefficients
scale	the scale parameter
sigma.e	the standard deviation of error terms
lk	the log-likelihood
npar	the total number of model parameters
aic	the AIC value
bic	the BIC value
qtl	the estimated quantile
nobs	the total number of observations
model	the estimated model
call	the matched call

summary.search_lqmix *Summary of a* search_lqmix *object*

Description

Summary method for the class search_lqmix.

Usage

```
## S3 method for class 'search_lqmix'
summary(object, ...)
```

Arguments

object	a search_lqmix object
...	not used

Value

Return an object of class summary.search_lqmix. This is a list of summary statistics for the optimal linear quantile mixture model given in object$optimal, with the following elements:

fix	a matrix with estimates, standard errors, Z statistics, and p-values for the fixed regression coefficients for the optimal fitted model
ranTC	a matrix with estimates, standard errors, Z statistics, and p-values for the TC random coefficients, if present for the optimal fitted model
ranTV	a matrix with estimates, standard errors, Z statistics, and p-values for the TV random coefficients, if present for the optimal fitted model
pg	a matrix with estimates and standard errors for the prior probabilities of the finite mixture associated to TC random coefficients, if present for the optimal fitted model
delta	a matrix with estimates and standard errors for the initial probabilities of the hidden Markov chain associated to TV random coefficients, if present for the optimal fitted model
Gamma	a matrix with estimates and standard errors for the transition probabilities of the hidden Markov chain associated to TV random coefficients, if present for the optimal fitted model
scale	the scale parameter for the optimal model
sigma.e	the standard deviation of error terms for the optimal model
logLik	the log-likelihood at convergence of the EM algorithm for the optimal model
npar	the total number of model parameters for the optimal model
AIC	the AIC value for the optimal model
BIC	the BIC value for the optimal model

qtl	the estimated quantile
G	the number of mixture components associated to TC random coefficients, if present for the optimal fitted model
m	the number of hidden states associated to TV random coefficients, if present for the optimal fitted model
nsbj	the number of subjects
nobs	the total number of observations
miss	the missingness type
model	the identified optimal model
call	the matched call

varAL *Variance of Asymmetric Laplace random variables*

Description

Compute the variance for the asymmetric Laplace Distribution.

Usage

```
varAL(sigma, qtl)
```

Arguments

sigma	scale parameter
qtl	skewness parameter

Value

Return the variance of Asymmetric Laplace random variables for given scale (sigma) and skewness (qtl) parameters.

References

Koenker R, Machado JAF (1999). "Goodness of fit and related inference processes for quantile regression." *Journal of the American Statistical Association*, **94**, 1296–1310.

Yu K, Moyeed RA (2001). "Bayesian quantile regression." *Statistics & Probability Letters*, **54**, 437–447.

Yu K, Zhang J (2005). "A three-parameter asymmetric Laplace distribution and its extension." *Communications in Statistics. Theory and Methods*, **34**, 1867–1879.

vcov.lqmix *Calculate variance-covariance matrix for a fitted* lqmix *object*

Description

Return the bootstrap variance-covariance matrix for the coefficients of a fitted lqmix object.

Usage

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

Arguments

object	an lqmix object
...	not used

Value

A matrix of the estimated covariances between the parameter estimates in the linear predictor of the model.

vcov.lqr *Calculate variance-covariance matrix for a fitted* lqr *object*

Description

Return the bootstrap variance-covariance matrix for the parameters of a fitted lqr object.

Usage

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

Arguments

object	an lqmix object
...	not used

Value

A matrix of the estimated covariances between the parameter estimates in the linear predictor of the model.

vcov.search_lqmix — *Calculate variance-covariance matrix for the optimal fitted model*

Description

Return the bootstrap variance-covariance matrix of the parameters of the optimal fitted model stored in an object of `class` `search_lqmix`.

Usage

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

Arguments

`object`	an `search_lqmix` object
`...`	not used

Value

A matrix of the estimated covariances between the main parameter estimates in the linear predictor of the model.

Index

∗ **datasets**
    cd4, 4
    pain, 13

cd4, 4
class, *5–8*, *14–22*, *24*, *26, 27*, *30*
coef.lqmix, 5
coef.lqr, 5
coef.search_lqmix, 6

dal, 6

logLik.lqmix, 7
logLik.lqr, 8
logLik.search_lqmix, 8
lqmix, *5*, *7, 8*, 9, *14, 15*, *17*, *19, 20*, *24*, *29*
lqmix-package, 2
lqr, *5*, *8*, 12, *16*, *18, 19*, *21*, *26*, *29*

pain, 13
plot.lqmix, 14
plot.search_lqmix, 15
predict.lqmix, 15
predict.lqr, 16
predict.search_lqmix, 16
print.lqmix, 17
print.lqr, 18
print.search_lqmix, 18
print.summary.lqmix, 19
print.summary.lqr, 19
print.summary.search_lqmix, 20

residuals.lqmix, 20
residuals.lqr, 21
residuals.search_lqmix, 22

search_lqmix, *6*, *8*, *15, 16*, *18*, *20*, *22*, 22, *27*, *30*
summary.lqmix, 24
summary.lqr, 26
summary.search_lqmix, 27

varAL, 28
vcov.lqmix, 29
vcov.lqr, 29
vcov.search_lqmix, 30