

Package: lqmix (via r-universe)

September 3, 2024

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 2023-11-07

Version 1.0

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, parallel

RdMacros Rdpack

RoxygenNote 7.2.3

NeedsCompilation no

Date/Publication 2023-12-07 10:50:11 UTC

Repository <https://mfrancesca-marino.r-universe.dev>

RemoteUrl <https://github.com/cran/lqmix>

RemoteRef HEAD

RemoteSha 4ea78c43139a17000b40fa14da6efb84c96e21fd

Contents

lqmix-package	2
cd4	3
coef.lqmix	4

coef.lqr	5
coef.search_lqmix	5
dal	6
logLik.lqmix	7
logLik.lqr	7
logLik.search_lqmix	8
lqmix	8
lqr	11
pain	12
plot.lqmix	13
plot.search_lqmix	14
print.lqmix	14
print.lqr	15
print.search_lqmix	15
print.summary.lqmix	16
print.summary.lqr	16
search_lqmix	17
summary.lqmix	19
summary.lqr	20
varAL	21

Index	22
--------------	-----------

lqmix-package	<i>Overview of the package lqmix</i>
---------------	--------------------------------------

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 unspecified 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 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 individual varies from 1 to 12. The CD4+ cell data are highly unbalanced.

References

Zeger, Scott L., and Peter J. Diggle. "Semiparametric models for longitudinal data with application to CD4 cell numbers in HIV seroconverters." *Biometrics* (1994): 689-699.

```
coef.lqmix
```

Print the estimated fixed coefficients of an lqmix object

Description

Print the estimated fixed coefficients of a fitted model of `class` lqmix

Usage

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

Arguments

```

object      an lqmix object
...         not used

```

Value

Return the estimated fixed coefficients obtained at convergence of the EM algorithm for a fitted model of `class` lqmix

coef.lqr	<i>Print the estimated fixed coefficients of an lqr object</i>
----------	--

Description

Print the estimated fixed coefficients of a fitted model of [class](#) lqr

Usage

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

Arguments

object	an lqmix object
...	not used

Value

Return the estimated coefficients obtained at convergence of the EM algorithm for a fitted model of [class](#) lqr

coef.search_lqmix	<i>Print the estimated fixed coefficients of the optimal model stored in a search_lqmix object</i>
-------------------	--

Description

Print the estimated fixed 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	an lqmix object
...	not used

Value

Return the estimated fixed coefficients obtained at convergence of the EM algorithm for the optimal model stored in an object 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

<code>y</code>	vector of quantiles
<code>mu</code>	location parameter
<code>sigma</code>	scale parameter
<code>qtl</code>	skewness parameter
<code>log</code>	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.

logLik.lqmix	<i>Print the log-likelihood of an lqmix object</i>
--------------	--

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	<i>Print the log-likelihood of an lqr object</i>
------------	--

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	<i>Print the log-likelihood of the optimal model stored in a search_lqmix object</i>
---------------------	--

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	<i>Linear Quantile Mixture with TC and/or TV, discrete, random coefficients</i>
-------	---

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 = 50, start = 0, parInit = list(betarf = NULL, betarTC = NULL, betarTV
      = NULL, pg = NULL, delta = NULL, Gamma = NULL, scale = NULL),
      verbose = TRUE, seed = NULL, parallel = FALSE)
```


Arguments

<code>formula</code>	an object of class <code>formula</code> : a symbolic description of the model to be fitted
<code>randomTC</code>	a one-sided formula of the form $\sim z_1 + z_2 + \dots + z_r$, where z_1, \dots, z_r denote the variables associated to TC random coefficients (1 for the intercept)
<code>randomTV</code>	a one-sided formula of the form $\sim w_1 + w_2 + \dots + w_l$, where w_1, \dots, w_l denote the variables associated to TV random coefficients (1 for the intercept). Note that only TC variables are allowed
<code>group</code>	a string indicating the grouping variable, i.e., the factor identifying the unit longitudinal measurements refer to
<code>time</code>	a string indicating the time variable
<code>G</code>	number of mixture components associated to TC random coefficients
<code>m</code>	number of states associated to the TV random coefficients
<code>data</code>	a data frame containing the variables named in <code>formula</code> , <code>randomTC</code> , <code>randomTV</code> , and <code>time</code>
<code>qtl</code>	quantile to be estimated
<code>eps</code>	tolerance level for (relative) convergence of the EM algorithm
<code>maxit</code>	maximum number of iterations for the EM algorithm
<code>se</code>	standard error computation for the optimal model
<code>R</code>	number of bootstrap samples for computing standard errors
<code>start</code>	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
<code>parInit</code>	list of initial model parameters when <code>start=2</code> . For a list of
<code>verbose</code>	if set to <code>FALSE</code> , no printed output is given during the function execution
<code>seed</code>	an integer value for random numbers generation
<code>parallel</code>	if set to <code>TRUE</code> , a parallelized code is use for standard error computation (if <code>se=TRUE</code>)

Details

The function computes ML estimates for the parameters of a linear quantile mixture model, based on TC and/or TV random coefficients. Estimates are derived by maximizing the (log-)likelihood of a Laplace regression where the location parameter is modeled as a function of fixed coefficients, together with TC and/or 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 admits the presence of missing data, both in terms of drop-outs (monotone missing data) and intermittent missing, 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:

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

References

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.

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 with no random coefficients

Usage

```
lqr(formula, data, qtl = 0.5, se = TRUE, R = 50, verbose = TRUE, ...)
```

Arguments

formula	an object of class <code>formula</code> : a symbolic description of the model to be fitted
data	a data frame containing the variables named in formula and time
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
...	further arguments to be passed to or from methods

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:

<code>betaf</code>	a vector containing fixed regression coefficients
<code>scale</code>	the scale parameter
<code>sigma.e</code>	the standard deviation of error terms
<code>lk</code>	the log-likelihood
<code>npar</code>	the total number of model parameters
<code>AIC</code>	the AIC value
<code>BIC</code>	the BIC value
<code>qt1</code>	the estimated quantile
<code>nobs</code>	the total number of observations
<code>se.betaf</code>	the standard errors for fixed regression coefficients
<code>se.scale</code>	the standard error for the scale parameter
<code>model</code>	the estimated model
<code>call</code>	the matched call

References

Geraci M, Bottai M (2007). "Quantile regression for longitudinal data using the asymmetric Laplace distribution." *Biostatistics*, **8**, 140-54.

Examples

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

`pain`

Pain Data

Description

The pain data frame is made by a total of 357 rows and 4 columns providing information on pain of 83 women in labor followed for a maximum of 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, Charles S. "Semi-parametric and non-parametric methods for the analysis of repeated measurements with applications to clinical trials." *Statistics in medicine* 10.12 (1991): 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 search_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 <code>search_lqmix</code>
...	not used

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

<code>x</code>	an <code>lqr</code> object
<code>digits</code>	a non-null value for digits specifying the minimum number of significant digits to be printed
<code>...</code>	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

<code>x</code>	a <code>search_lqmix</code> object
<code>digits</code>	a non-null value for digits specifying the minimum number of significant digits to be printed
<code>...</code>	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

<code>x</code>	a summary of an <code>lqmix</code> object
<code>digits</code>	a non-null value for digits specifying the minimum number of significant digits to be printed
<code>...</code>	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

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

Value

Return a summary of an `lqr` object

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 = 50, verbose = TRUE,
             seed = NULL, parallel = FALSE)
```

Arguments

formula	an object of <code>class</code> formula: a symbolic description of the model to be fitted
randomTC	a one-sided formula of the form $\sim z_1 + z_2 + \dots + z_r$, where z_1, \dots, z_r denote the variables associated to TC random coefficients (1 for the intercept)
randomTV	a one-sided formula of the form $\sim w_1 + w_2 + \dots + w_l$, where w_1, \dots, w_l 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)
mv	vector of possible number of states associated to the TV random coefficients (if present)
data	a data frame containing the variables named in formula, randomTC, randomTV, 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 (relative) 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
parallel	if set to TRUE, a parallelized code is use for standard error computation (if se=TRUE)

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 of random tries is proportional to the number of mixture components and/or hidden states associated to the random coefficients in the model.

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:

<code>optimal</code>	the identified optimal model
<code>allmodels</code>	the output of each estimated model
<code>lkv</code>	the vector of likelihood values for each estimated model
<code>aicv</code>	the vector of AIC values for each estimated model
<code>bicv</code>	the vector of BIC values for each estimated model
<code>qt1</code>	the estimated quantile
<code>mv</code>	the vector of possible number of states associated to TV random coefficients (if present)
<code>Gv</code>	the vector of possible number of mixture components associated to TC random coefficients (if present)
<code>method</code>	the method used to select the optimal model
<code>call</code>	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
qt1	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)
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

`varAL`*Variance of Asymmetric Laplace random variables*

Description

Compute the variance for the asymmetric Laplace distribution

Usage

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

Arguments

<code>sigma</code>	scale parameter
<code>qtl</code>	skewness parameter

Value

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

Index

* datasets

cd4, 3
pain, 12

cd4, 3
class, 4, 5, 7, 8, 10, 12–20
coef.lqmix, 4
coef.lqr, 5
coef.search_lqmix, 5

dal, 6

formula, 9, 11

logLik.lqmix, 7
logLik.lqr, 7
logLik.search_lqmix, 8
lqmix, 8
lqmix-package, 2
lqr, 11

pain, 12
plot.lqmix, 13
plot.search_lqmix, 14
print.lqmix, 14
print.lqr, 15
print.search_lqmix, 15
print.summary.lqmix, 16
print.summary.lqr, 16

search_lqmix, 13, 14, 17
summary.lqmix, 19
summary.lqr, 20

varAL, 21