Package 'lqr'

July 22, 2025

Type Package

Title Robust Linear Quantile Regression

Version 5.2

Date 2024-07-12

Author Christian E. Galarza <chedgala@espol.edu.ec>, Luis Benites <lbenitess@pucp.edu.pe>, Marcelo Bourguignon <m.p.bourguignon@gmail.com>, Victor H. Lachos <hlachos@uconn.edu>

Maintainer Christian E. Galarza <cgalarza88@gmail.com>

Imports graphics, stats, spatstat.univar, numDeriv, MomTrunc, quantreg, MASS

Suggests ald

Description It fits a robust linear quantile regression model using a new family of zero-quantile distributions for the error term. Missing values and censored observations can be handled as well. This family of distribution includes skewed versions of the Normal, Student's t, Laplace, Slash and Contaminated Normal distribution. It also performs logistic quantile regression for bounded responses as shown in Galarza et.al.(2020) <doi:10.1007/s13571-020-00231-0>. It provides estimates and full inference. It also provides envelopes plots for assessing the fit and confidences bands when several quantiles are provided simultaneously.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2024-07-12 20:10:01 UTC

Contents

lqr-package 2
ais 3
best.lqr 3
cens.lqr 5
dist.Truncated 8
Egig 9
Log.best.lqr 10

Log.lqr 13
lqr 18
resistance 23
SKD 23

Index **26**

`lqr-package` *Robust Linear Quantile Regression*

Description

It fits a robust linear quantile regression model using a new family of zero-quantile distributions for the error term. This family of distribution includes skewed versions of the Normal, Student's t, Laplace, Slash and Contaminated Normal distribution. It provides estimates and full inference. It also provides envelopes plots for assessing the fit and confidences bands when several quantiles are provided simultaneously. Details of its first version can be found below.

Author(s)

Christian E. Galarza <<cgalarza88@gmail.com>>, Luis Benites <<lsanchez@ime.usp.br>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<cgalarza88@gmail.com>>

References

Galarza, C., Lachos, V. H. & Bourguignon M. (2021). A skew-t quantile regression for censored and missing data. Stat.<doi:10.1002/sta4.379>.

Galarza C.E., Lachos V.H. & Panpan Z. (2020) Logistic quantile regression for bounded outcomes using a family of heavy-tailed distributions. Sankhya B. <doi:10.1007/s13571-020-00231-0>.

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

See Also

`SKD`,`Log.best.lqr`, `Log.lqr`,`best.lqr`,`lqr`, `ais`

ais *Australian institute of sport data*

Description

Data on 102 male and 100 female athletes collected at the Australian Institute of Sport.

Format

This data frame contains the following columns:

Sex (0 = male or 1 = female)

Ht height (cm)

Wt weight (kg)

LBM lean body mass

RCC red cell count

WCC white cell count

Hc Hematocrit

Hg Hemoglobin

Ferr plasma ferritin concentration

BMI body mass index, weight/(height)**2

SSF sum of skin folds

Bfat Percent body fat

Label Case Labels

Sport Sport

References

S. Weisberg (2005). *Applied Linear Regression*, 3rd edition. New York: Wiley, Section 6.4

best.lqr *Best Fit in Robust Linear Quantile Regression*

Description

It finds the best fit distribution in robust linear quantile regression model. It adjusts the Normal, Student's t, Laplace, Slash and Contaminated Normal models. It shows a summary table with the likelihood-based criterion, envelopes plots and the histogram of the residuals with fitted densities for all models. Estimates and full inference are provided for the best model.

Usage

```
best.lqr(formula,data = NULL,subset = NULL,
             p = 0.5, precision = 10^-6,
             criterion = "AIC")
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula).
subset	an optional string specifying a subset of observations to be used in the fitting process. Be aware of the use of double quotes in a proper way when necessary, e.g., in "(sex=='F')".
p	An unique quantile or a set of quantiles related to the quantile regression.
precision	The convergence maximum error permitted. By default is 10^-6.
criterion	Likelihood-based criterion to be used for choosen the best model. It could be AIC, BIC, HQ or loglik (log-likelihood). By default AIC criterion will be used.

Details

The best.fit() function finds the best model only for one quantile. For fitting a grid of quantiles lqr() might be used but the distribution must be provided.

Value

For the best model:

iter	number of iterations.
criteria	attained criteria value.
beta	fixed effects estimates.
sigma	scale parameter estimate for the error term.
nu	Estimate of nu parameter detailed above.
gamma	Estimate of gamma parameter detailed above.
SE	Standard Error estimates.
table	Table containing the inference for the fixed effects parameters.
loglik	Log-likelihood value.
AIC	Akaike information criterion.
BIC	Bayesian information criterion.
HQ	Hannan-Quinn information criterion.
fitted.values	vector containing the fitted values.
residuals	vector containing the residuals.

Author(s)

Christian E. Galarza <<cgalarza88@gmail.com>>, Luis Benites <<lsanchez@ime.usp.br>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<cgalarza88@gmail.com>>

References

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

Wichitaksorn, N., Choy, S. T., & Gerlach, R. (2014). A generalized class of skew distributions and associated robust quantile regression models. Canadian Journal of Statistics, 42(4), 579-596.

See Also

lqr,Log.lqr,Log.best.lqr,dSKD

Examples

```
data(crabs,package = "MASS")

#Finding the best model for the 3rd quartile based on BIC
best.lqr(BD~FL,data = crabs, p = 0.75, criterion = "BIC")
```

---

cens.lqr *Skew-t quantile regression for censored and missing data*

Description

It fits a linear quantile regression model where the error term is considered to follow an SKT skew-t distribution, that is, the one proposed by Wichitaksorn et.al. (2014). Additionally, the model is capable to deal with missing and interval-censored data at the same time. Degrees of freedom can be either estimated or supplied by the user. It offers estimates and full inference. It also provides envelopes plots and likelihood-based criteria for assessing the fit, as well as fitted and imputed values.

Usage

```
cens.lqr(y,x,cc,LL,UL,p=0.5,nu=NULL,precision=1e-06,envelope=FALSE)
```

Arguments

y	the response vector of dimension n where n is the total of observations. It may contain both missing and censored values represented by NaNs.
x	design matrix for the fixed effects of dimension Nxd where d represents the number of fixed effects including the intercept, if considered.
cc	vector of censoring/missing indicators. For each observation it takes 0 if non-censored/missing, 1 if censored/missing.
LL	the vector of lower limits of dimension nx1. See details section.
UL	the vector of upper limits of dimension nx1. See details section.
p	An unique quantile of interest to fit the quantile regression.
nu	It represents the degrees of freedom of the skew-t distribution. When is not provided, we use the MLE.
precision	The convergence maximum error permitted. By default is 10^-6.
envelope	if TRUE, it will show a confidence envelope for a curve based on bootstrap replicates. it is FALSE by default.

Details

Missing or censored values in the response can be represented imputed as NaNs, since the algorithm only uses the information provided in the lower and upper limits LL and UL. The indicator vector cc must take the value of 1 for these observations.

Censored and missing data

If all lower limits are -Inf, we will be dealing with left-censored data. Besides, if all upper limits are Inf, this is the case of right-censored data. Interval-censoring is considered when both limits are finites. If some observation is missing, we have not information at all, so both limits must be infinites.

Combinations of all cases above are permitted, that is, we may have left-censored, right-censored, interval-censored and missing data at the same time.

Value

iter	number of iterations.
criteria	attained criteria value.
beta	fixed effects estimates.
sigma	scale parameter estimate for the error term.
nu	Estimate of nu parameter detailed above.
SE	Standard Error estimates.
table	Table containing the inference for the fixed effects parameters.
loglik	Log-likelihood value.
AIC	Akaike information criterion.
BIC	Bayesian information criterion.
HQ	Hannan-Quinn information criterion.

`fitted.values`	vector containing the fitted values.
`imputed.values`	vector containing the imputed values for censored/missing observations.
`residuals`	vector containing the residuals.

Author(s)

Christian E. Galarza <<chedgala@espol.edu.ec>>, Marcelo Bourguignon <<m.p.bourguignon@gmail.com>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<chedgala@espol.edu.ec>>

References

Galarza, C., Lachos, V. H. & Bourguignon M. (2021). A skew-t quantile regression for censored and missing data. Stat.doi:10.1002/sta4.379.

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

Wichitaksorn, N., Choy, S. B., & Gerlach, R. (2014). A generalized class of skew distributions and associated robust quantile regression models. Canadian Journal of Statistics, 42(4), 579-596.

See Also

`lqr`,`best.lqr`,`Log.lqr`, `Log.best.lqr`,`dSKD`

Examples

```
##Load the data
data(ais)
attach(ais)

##Setting
y<-BMI
x<-cbind(1,LBM,Sex)

cc = rep(0,length(y))
LL = UL = rep(NA,length(y))

#Generating a 5% of interval-censored values
ind = sample(x = c(0,1),size = length(y),
replace = TRUE,prob = c(0.95,0.05))
ind1 = (ind == 1)

cc[ind1] = 1
LL[ind1] = y[ind1] - 10
UL[ind1] = y[ind1] + 10
y[ind1] = NA #deleting data

#Fitting the model

# A median regression with unknown degrees of freedom
out = cens.lqr(y,x,cc,LL,UL,p=0.5,nu = NULL,precision = 1e-6,envelope = TRUE)
```

```
# A first quartile regression with 10 degrees of freedom
out = cens.lqr(y,x,cc,LL,UL,p=0.25,nu = 10,precision = 1e-6,envelope = TRUE)
```

`dist.Truncated` *Truncated Distributions*

Description

Density, distribution function, quantile function and random generation for truncated distributions.

Usage

```
dtrunc(x, spec, a=-Inf, b=Inf, log=FALSE, ...)
extrunc(spec, a=-Inf, b=Inf, ...)
ptrunc(x, spec, a=-Inf, b=Inf, ...)
qtrunc(p, spec, a=-Inf, b=Inf, ...)
rtrunc(n, spec, a=-Inf, b=Inf, ...)
vartrunc(spec, a=-Inf, b=Inf, ...)
```

Arguments

`n`	This is a the number of random draws for `rtrunc`.
`p`	This is a vector of probabilities.
`x`	This is a vector to be evaluated.
`spec`	The base name of a probability distribution is specified here. For example, to estimate the density of a truncated normal distribution, enter `norm`.
`a`	This is the lower bound of truncation, which defaults to negative infinity.
`b`	This is the upper bound of truncation, which defaults to infinity.
`log`	Logical. If `log=TRUE`, then the logarithm of the density is returned.
`...`	Additional arguments to pass.

Details

A truncated distribution is a conditional distribution that results from a priori restricting the domain of some other probability distribution. More than merely preventing values outside of truncated bounds, a proper truncated distribution integrates to one within the truncated bounds. In contrast to a truncated distribution, a censored distribution occurs when the probability distribution is still allowed outside of a pre-specified range. Here, distributions are truncated to the interval $[a, b]$, such as $p(\theta) \in [a, b]$.

The R code of Nadarajah and Kotz (2006) has been modified to work with log-densities. This code was also available in the (extinct) package LaplacesDemon.

Value

dtrunc gives the density, extrunc gives the expectation, ptrunc gives the distribution function, qtrunc gives the quantile function, rtrunc generates random deviates, and vartrunc gives the variance of the truncated distribution.

References

Nadarajah, S. and Kotz, S. (2006). "R Programs for Computing Truncated Distributions". *Journal of Statistical Software*, 16, Code Snippet 2, p. 1–8.

See Also

lqr, SKD.

Examples

```
x <- seq(-0.5, 0.5, by = 0.1)
y <- dtrunc(x, "norm", a=-0.5, b=0.5, mean=0, sd=2)
```

Egig *Moments of the Generalized Inverse Gaussian Distribution*

Description

Expected value of X, log(X), 1/X and variance for the generalized inverse gaussian distribution. This function has been recycled from the ghyp R package.

Usage

```
Egig(lambda, chi, psi, func = c("x", "logx", "1/x", "var"))
```

Arguments

lambda	A shape and scale and parameter.
chi, psi	Shape and scale parameters. Must be positive.
func	The transformation function when computing the expected value. x is the expected value (default), log x returns the expected value of the logarithm of x, 1/x returns the expected value of the inverse of x and var returns the variance.

Details

Egig with func = "log x" uses grad from the R package *numDeriv*. See the package vignette for details regarding the expectation of GIG random variables.

Value

Egig gives the expected value of either x, 1/x, log(x) or the variance if func equals var.

Author(s)

David Luethi and Ester Pantaleo

References

Dagpunar, J.S. (1989). *An easily implemented generalised inverse Gaussian generator.* Commun. Statist. -Simula., **18**, 703–710.

Michael, J. R, Schucany, W. R, Haas, R, W. (1976). *Generating random variates using transformations with multiple roots*, The American Statistican, **30**, 88–90.

See Also

best.lqr

Examples

```
Egig(lambda = 10, chi = 1, psi = 1, func = "x")
Egig(lambda = 10, chi = 1, psi = 1, func = "var")
Egig(lambda = 10, chi = 1, psi = 1, func = "1/x")
```

Log.best.lqr *Best Fit in Robust Logistic Linear Quantile Regression*

Description

It performs the logistic transformation in Galarza et.al.(2020) (see references) for estimating quantiles for a bounded response. Once the response is transformed, it uses the best.lqr function.

Usage

```
Log.best.lqr(formula,data = NULL,subset = NULL,
                        p=0.5,a=0,b=1,
                        epsilon = 0.001,precision = 10^-6,
                        criterion = "AIC")
```

Arguments

We will detail first the only three arguments that differ from best.lqr function.

a	lower bound for the response (default = 0)
b	upper bound for the response (default = 1)
epsilon	a small quantity $\epsilon > 0$ that ensures that the logistic transform is defined for all values of the response.
formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.

data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula).
subset	an optional string specifying a subset of observations to be used in the fitting process. Be aware of the use of double quotes in a proper way when necessary, e.g., in "(sex=='F')".
p	An unique quantile or a set of quantiles related to the quantile regression.
precision	The convergence maximum error permitted. By default is 10^-6.
criterion	Likelihood-based criterion to be used for choosen the best model. It could be AIC, BIC, HQ or loglik (log-likelihood). By default AIC criterion will be used.

Details

We follow the transformation in Bottai et.al. (2009) defined as

$$h(y) = logit(y) = log(\frac{y-a}{b-y})$$

that implies

$$Q_y(p) = \frac{b\, exp(X\beta) + a}{1 + exp(X\beta)}$$

where $Q_y(p)$ represents the conditional quantile of the response. Once estimates for the regression coefficients β_p are obtained, inference on $Q_y(p)$ can then be made through the inverse transform above. This equation (as function) is provided in the output. See example.

The interpretation of the regression coefficients is analogous to the interpretation of the coefficients of a logistic regression for binary outcomes.

For example, let x_1 be the gender (male = 0, female=1). Then $exp(\beta_{0.5,1})$ represents the odds ratio of median score in males vs females, where the odds are defined using the score instead of a probability, $(y-a)/(b-y)$. When the covariate is continous, the respective β coeficient can be interpretated as the increment (or decrement) over the log(odd ratio) when the covariate increases one unit.

Value

For the best model:

iter	number of iterations.
criteria	attained criteria value.
beta	fixed effects estimates.
sigma	scale parameter estimate for the error term.
nu	Estimate of nu parameter detailed above.
gamma	Estimate of gamma parameter detailed above.
SE	Standard Error estimates.
table	Table containing the inference for the fixed effects parameters.

loglik	Log-likelihood value.
AIC	Akaike information criterion.
BIC	Bayesian information criterion.
HQ	Hannan-Quinn information criterion.
fitted.values	vector containing the fitted values.
residuals	vector containing the residuals.

Note

When a grid of quantiles is provided, a graphical summary with point estimates and Confidence Intervals for model parameters is shown. Also, the result will be a list of the same dimension where each element corresponds to each quantile as detailed above.

Author(s)

Christian E. Galarza <<cgalarza88@gmail.com>>, Luis Benites <<lsanchez@ime.usp.br>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<cgalarza88@gmail.com>>

References

Galarza, C.M., Zhang P. and Lachos, V.H. (2020). Logistic Quantile Regression for Bounded Outcomes Using a Family of Heavy-Tailed Distributions. Sankhya B: The Indian Journal of Statistics. doi:10.1007/s13571020002310

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

See Also

Log.lqr,best.lqr,dSKD

Examples

```
##Load the data
data(resistance)
attach(resistance)

#EXAMPLE 1.1

#Comparing the resistence to death of two types of tumor-cells.
#The response is a score in [0,4].

boxplot(score~type)

#Median logistic quantile regression (Best fit distribution)
res = Log.best.lqr(formula = score~type,data = resistance,a=0,b=4)

# The odds ratio of median score in type B vs type A
```

```
exp(res$beta[2])

#Proving that exp(res$beta[2])  is approx median odd ratio
medA  = median(score[type=="A"])
medB  = median(score[type=="B"])
rateA = (medA - 0)/(4 - medA)
rateB = (medB - 0)/(4 - medB)
odd   = rateB/rateA

round(c(exp(res$beta[2]),odd),3) #best fit

#EXAMPLE 1.2
#############

#Comparing the resistence to death depending of dose.

#descriptive
plot(dose,score,ylim=c(0,4),col="dark gray");abline(h=c(0,4),lty=2)
dosecat<-cut(dose, 6, ordered = TRUE)
boxplot(score~dosecat,ylim=c(0,4))
abline(h=c(0,4),lty=2)

#(Non logistic) Best quantile regression for quantiles
# 0.05, 0.50 and 0.95
p05 = best.lqr(score~poly(dose,3),data = resistance,p = 0.05)
p50 = best.lqr(score~poly(dose,3),data = resistance,p = 0.50)
p95 = best.lqr(score~poly(dose,3),data = resistance,p = 0.95)
res3  = list(p05,p50,p95)

plot(dose,score,ylim=c(-1,5),col="gray");abline(h=c(0,4),lty=2)
lines(sort(dose), p05$fitted.values[order(dose)], col='red', type='l')
lines(sort(dose), p50$fitted.values[order(dose)], col='blue', type='l')
lines(sort(dose), p95$fitted.values[order(dose)], col='red', type='l')

#Using logistic quantile regression for obtaining predictions inside bounds

logp05 = Log.best.lqr(score~poly(dose,3),data = resistance,p = 0.05,b = 4) #a = 0 by default
logp50 = Log.best.lqr(score~poly(dose,3),data = resistance,p = 0.50,b = 4)
logp95 = Log.best.lqr(score~poly(dose,3),data = resistance,p = 0.95,b = 4)
res4  = list(logp05,logp50,logp95)

#No more prediction curves out-of-bounds
plot(dose,score,ylim=c(-1,5),col="gray");abline(h=c(0,4),lty=2)
lines(sort(dose), logp05$fitted.values[order(dose)], col='red', type='l')
lines(sort(dose), logp50$fitted.values[order(dose)], col='blue', type='l')
lines(sort(dose), logp95$fitted.values[order(dose)], col='red', type='l')
```

Log.lqr *Robust Logistic Linear Quantile Regression*

Description

It performs the logistic transformation in Galarza et.al.(2020) (see references) for estimating quantiles for a bounded response. Once the response is transformed, it uses the `lqr` function.

Usage

```
Log.lqr(formula,data = NULL,subset = NULL,
                    p=0.5,a=0,b=1,
                    dist = "normal",
                    nu=NULL,
                    gamma=NULL,
                    precision = 10^-6,
                    epsilon = 0.001,
                    CI=0.95,
                    silent = FALSE)
```

Arguments

We will detail first the only three arguments that differ from `lqr` function.

`a`	lower bound for the response (default = 0)
`b`	upper bound for the response (default = 1)
`epsilon`	a small quantity $\epsilon > 0$ that ensures that the logistic transform is defined for all values of the response.
`formula`	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in `data`, the variables are taken from `environment(formula)`.
`subset`	an optional string specifying a subset of observations to be used in the fitting process. Be aware of the use of double quotes in a proper way when necessary, e.g., in `"(sex=='F')"`.
`p`	An unique quantile or a set of quantiles related to the quantile regression.
`dist`	represents the distribution to be used for the error term. The values are `normal` for Normal distribution, `t` for Student's t distribution, `laplace` for Laplace distribution, `slash` for Slash distribution and `cont` for the Contaminated normal distribution.
`nu`	It represents the degrees of freedom when `dist = t`. For the Slash distribution (`dist = slash`) it is a shape parameter $\nu > 0$. For the Contaminated Normal distribution, ν is the parameter that represents the percentage of outliers. When is not provided, we use the MLE.
`gamma`	It represents a scale factor for the contaminated normal distribution. When is not provided, we use the MLE.
`precision`	The convergence maximum error permitted. By default is 10^-6.
`CI`	Confidence to be used for the Confidence Interval when a grid of quantiles is provided. Default = 0.95.
`silent`	if `FALSE` (by default), the function prints some output.

Details

We follow the transformation in Bottai et.al. (2009) defined as

$$h(y) = logit(y) = log(\frac{y-a}{b-y})$$

that implies

$$Q_y(p) = \frac{b\, exp(X\beta) + a}{1 + exp(X\beta)}$$

where $Q_y(p)$ represents the conditional quantile of the response. Once estimates for the regression coefficients β_p are obtained, inference on $Q_y(p)$ can then be made through the inverse transform above. This equation (as function) is provided in the output. See example.

The interpretation of the regression coefficients is analogous to the interpretation of the coefficients of a logistic regression for binary outcomes.

For example, let x_1 be the gender (male = 0, female=1). Then $exp(\beta_{0.5,1})$ represents the odds ratio of median score in males vs females, where the odds are defined using the score instead of a probability, $(y-a)/(b-y)$. When the covariate is continous, the respective β coeficient can be interpretated as the increment (or decrement) over the log(odd ratio) when the covariate increases one unit.

Value

`iter`	number of iterations.
`criteria`	attained criteria value.
`beta`	fixed effects estimates.
`sigma`	scale parameter estimate for the error term.
`nu`	Estimate of `nu` parameter detailed above.
`gamma`	Estimate of `gamma` parameter detailed above.
`SE`	Standard Error estimates.
`table`	Table containing the inference for the fixed effects parameters.
`loglik`	Log-likelihood value.
`AIC`	Akaike information criterion.
`BIC`	Bayesian information criterion.
`HQ`	Hannan-Quinn information criterion.
`fitted.values`	vector containing the fitted values.
`residuals`	vector containing the residuals.

Note

When a grid of quantiles is provided, a graphical summary with point estimates and Confidence Intervals for model parameters is shown. Also, the result will be a list of the same dimension where each element corresponds to each quantile as detailed above.

Author(s)

Christian E. Galarza <<cgalarza88@gmail.com>>, Luis Benites <<lsanchez@ime.usp.br>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<cgalarza88@gmail.com>>

References

Galarza, C.M., Zhang P. and Lachos, V.H. (2020). Logistic Quantile Regression for Bounded Outcomes Using a Family of Heavy-Tailed Distributions. Sankhya B: The Indian Journal of Statistics. doi:10.1007/s13571020002310

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

See Also

Log.best.lqr,best.lqr,dSKD

Examples

```
##Load the data
data(resistance)
attach(resistance)

#EXAMPLE 1.1

#Comparing the resistence to death of two types of tumor-cells.
#The response is a score in [0,4].

boxplot(score~type,ylab="score",xlab="type")

#Student't median logistic quantile regression
res = Log.lqr(score~type,data = resistance,a=0,b=4,dist="t")

# The odds ratio of median score in type B vs type A
exp(res$beta[2])

#Proving that exp(res$beta[2])  is approx median odd ratio
medA  = median(score[type=="A"])
medB  = median(score[type=="B"])
rateA = (medA - 0)/(4 - medA)
rateB = (medB - 0)/(4 - medB)
odd   = rateB/rateA

round(c(exp(res$beta[2]),odd),3)


#EXAMPLE 1.2
#############

#Comparing the resistence to death depending of dose.
```

```
#descriptive
plot(dose,score,ylim=c(0,4),col="dark gray");abline(h=c(0,4),lty=2)
dosecat<-cut(dose, 6, ordered = TRUE)
boxplot(score~dosecat,ylim=c(0,4))
abline(h=c(0,4),lty=2)

#(Non logistic) Best quantile regression for quantiles
# 0.05, 0.50 and 0.95
p05 = best.lqr(score~poly(dose,3),data = resistance,p = 0.05)
p50 = best.lqr(score~poly(dose,3),data = resistance,p = 0.50)
p95 = best.lqr(score~poly(dose,3),data = resistance,p = 0.95)
res3  = list(p05,p50,p95)

plot(dose,score,ylim=c(-1,5),col="gray");abline(h=c(0,4),lty=2)
lines(sort(dose), p05$fitted.values[order(dose)], col='red', type='l')
lines(sort(dose), p50$fitted.values[order(dose)], col='blue', type='l')
lines(sort(dose), p95$fitted.values[order(dose)], col='red', type='l')

#Using Student's t logistic quantile regression for obtaining preditypeBions inside bounds

logp05 = Log.lqr(score~poly(dose,3),data = resistance,p = 0.05,b = 4,dist = "t") #a = 0 by default
logp50 = Log.lqr(score~poly(dose,3),data = resistance,p = 0.50,b = 4,dist = "t")
logp95 = Log.lqr(score~poly(dose,3),data = resistance,p = 0.95,b = 4,dist = "t")
res4  = list(logp05,logp50,logp95)

#No more predited curves out-of-bounds
plot(dose,score,ylim=c(-1,5),col="gray");abline(h=c(0,4),lty=2)
lines(sort(dose), logp05$fitted.values[order(dose)], col='red', type='l')
lines(sort(dose), logp50$fitted.values[order(dose)], col='blue', type='l')
lines(sort(dose), logp95$fitted.values[order(dose)], col='red', type='l')

#EXAMPLE 1.3
#############

#A full model using dose and type for a grid of quantiles

res5 = Log.lqr(formula = score ~ poly(dose,3)*type,data = resistance,
               a = 0,b = 4,
               p = seq(from = 0.05,to = 0.95,by = 0.05),dist = "t",
               silent = TRUE)

#A nice plot

if(TRUE){
  par(mfrow=c(1,2))
  typeB = (resistance$type == "B")

  plot(dose,score,
       ylim=c(0,4),
       col=c(8*typeB + 1*!typeB),main="Type A")
  abline(h=c(0,4),lty=2)
```

```
  lines(sort(dose[!typeB]),
        res5[[2]]$fitted.values[!typeB][order(dose[!typeB])],
        col='red')

  lines(sort(dose[!typeB]),
        res5[[5]]$fitted.values[!typeB][order(dose[!typeB])],
        col='green')

  lines(sort(dose[!typeB]),
        res5[[10]]$fitted.values[!typeB][order(dose[!typeB])],
        col='blue',lwd=2)

  lines(sort(dose[!typeB]),
        res5[[15]]$fitted.values[!typeB][order(dose[!typeB])],
        col='green')

  lines(sort(dose[!typeB]),
        res5[[18]]$fitted.values[!typeB][order(dose[!typeB])],
        col='red')

  plot(dose,score,
       ylim=c(0,4),
       col=c(1*typeB + 8*!typeB),main="Type B")
  abline(h=c(0,4),lty=2)

  lines(sort(dose[typeB]),
        res5[[2]]$fitted.values[typeB][order(dose[typeB])],
        col='red')

  lines(sort(dose[typeB]),
        res5[[5]]$fitted.values[typeB][order(dose[typeB])],
        col='green')

  lines(sort(dose[typeB]),
        res5[[10]]$fitted.values[typeB][order(dose[typeB])],
        col='blue',lwd=2)

  lines(sort(dose[typeB]),
        res5[[15]]$fitted.values[typeB][order(dose[typeB])],
        col='green')

  lines(sort(dose[typeB]),
        res5[[18]]$fitted.values[typeB][order(dose[typeB])],
        col='red')
}
```

---

lqr *Robust Linear Quantile Regression*

Description

It fits a robust linear quantile regression model using a new family of zero-quantile distributions for the error term. This family of distribution includes skewed versions of the Normal, Student's t, Laplace, Slash and Contaminated Normal distribution. It provides estimates and full inference. It also provides envelopes plots for assessing the fit and confidences bands when several quantiles are provided simultaneously.

Usage

```
lqr(formula,data = NULL,subset = NULL,
               p=0.5,dist = "normal",
               nu=NULL,gamma=NULL,
               precision = 10^-6,envelope=FALSE,
               CI=0.95,silent = FALSE
)

#lqr(y~x, data, p = 0.5, dist = "normal")
#lqr(y~x, data, p = 0.5, dist = "t")
#lqr(y~x, data, p = 0.5, dist = "laplace")
#lqr(y~x, data, p = 0.5, dist = "slash")
#lqr(y~x, data, p = 0.5, dist = "cont")

#lqr(y~x, p = c(0.25,0.50,0.75), dist = "normal")
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula).
subset	an optional string specifying a subset of observations to be used in the fitting process. Be aware of the use of double quotes in a proper way when necessary, e.g., in "(sex=='F')".
p	An unique quantile or a set of quantiles related to the quantile regression.
dist	represents the distribution to be used for the error term. The values are normal for Normal distribution, t for Student's t distribution, laplace for Laplace distribution, slash for Slash distribution and cont for the Contaminated normal distribution.
nu	It represents the degrees of freedom when dist = t. For the Slash distribution (dist = slash) it is a shape parameter $\nu > 0$. For the Contaminated Normal distribution, ν is the parameter that represents the percentage of outliers. When is not provided, we use the MLE.
gamma	It represents a scale factor for the contaminated normal distribution. When is not provided, we use the MLE.
precision	The convergence maximum error permitted. By default is 10^-6.

envelope	if TRUE, it will show a confidence envelope for a curve based on bootstrap replicates. By default it is TRUE when just one quantile is provided. If a grid of quantiles is provided it will be FALSE by default.
CI	Confidence to be used for the Confidence Interval when a grid of quantiles is provided. Default = 0.95.
silent	if FALSE (by default), the function prints some output.

Details

When a grid of quantiles is provided, a graphical summary with point estimates and Confidence Intervals for model parameters is shown.

Value

iter	number of iterations.
criteria	attained criteria value.
beta	fixed effects estimates.
sigma	scale parameter estimate for the error term.
nu	Estimate of nu parameter detailed above.
gamma	Estimate of gamma parameter detailed above.
SE	Standard Error estimates.
table	Table containing the inference for the fixed effects parameters.
loglik	Log-likelihood value.
AIC	Akaike information criterion.
BIC	Bayesian information criterion.
HQ	Hannan-Quinn information criterion.
fitted.values	vector containing the fitted values.
residuals	vector containing the residuals.

Note

If a grid of quantiles is provided, the result will be a list of the same dimension where each element corresponds to each quantile as detailed above.

Author(s)

Christian E. Galarza <<cgalarza88@gmail.com>>, Luis Benites <<lsanchez@ime.usp.br>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<cgalarza88@gmail.com>>

References

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

Wichitaksorn, N., Choy, S. T., & Gerlach, R. (2014). A generalized class of skew distributions and associated robust quantile regression models. Canadian Journal of Statistics, 42(4), 579-596.

See Also

cens.lqr,best.lqr,Log.lqr, Log.best.lqr,dSKD

Examples

```
#Example 1

##Load the data
data(ais)
attach(ais)

## Fitting a median regression with Normal errors (by default)

modelF = lqr(BMI~LBM,data = ais,subset = "(Sex==1)")
modelM = lqr(BMI~LBM,data = ais,subset = "(Sex==0)")

plot(LBM,BMI,col=Sex*2+1,
     xlab="Lean Body Mass",
     ylab="Body4 Mass Index",
     main="Quantile Regression")

abline(a = modelF$beta[1],b = modelF$beta[2],lwd=2,col=3)
abline(a = modelM$beta[1],b = modelM$beta[2],lwd=2,col=1)
legend(x = "topleft",legend = c("Male","Female"),lwd = 2,col = c(1,3))

#COMPARING SOME MODELS for median regression
modelN  = lqr(BMI~LBM,dist = "normal")
modelT  = lqr(BMI~LBM,dist = "t")
modelL  = lqr(BMI~LBM,dist = "laplace")

#Comparing AIC criteria
modelN$AIC;modelT$AIC;modelL$AIC

#This could be automatically done using best.lqr()
best.model = best.lqr(BMI~LBM,data = ais,
                      p = 0.75, #third quartile
                      criterion = "AIC")

#Let's use a grid of quantiles (no output)
modelfull = lqr(BMI~LBM,data = ais,
                p = seq(from = 0.10,to = 0.90,by = 0.05),
                dist = "normal",silent = TRUE)

#Plotting quantiles 0.10,0.25,0.50,0.75 and 0.90

if(TRUE){
  plot(LBM,BMI,xlab = "Lean Body Mass"
       ,ylab = "Body Mass Index", main = "Quantile Regression",pch=16)

  colvec = c(2,2,3,3,4)
  imodel = c(1,17,4,14,9)
  for(i in 1:5){
```

```
    abline(a = modelfull[[imodel[i]]]$beta[1],
           b = modelfull[[imodel[i]]]$beta[2],
           lwd=2,col=colvec[i])
  }
  legend(x = "topleft",
         legend = rev(c("0.10","0.25","0.50","0.75","0.90")),
         lwd = 2,col = c(2,3,4,3,2))
}

#Example 2
##Load the data

data(crabs,package = "MASS")
attach(crabs)

## Fitting a median regression with Normal errors (by default) #Note the double quotes
crabsF = lqr(BD~FL,data = crabs,subset = "(sex=='F')")
crabsM = lqr(BD~FL,data = crabs,subset = "(sex=='M')")

if(TRUE){
  plot(FL,BD,col=as.numeric(sex)+1,
       xlab="Frontal lobe size",ylab="Body depth",main="Quantile Regression")
  abline(a = crabsF$beta[1],b = crabsF$beta[2],lwd=2,col=2)
  abline(a = crabsM$beta[1],b = crabsM$beta[2],lwd=2,col=3)
  legend(x = "topleft",legend = c("Male","Female"),
         lwd = 2,col = c(3,2))
}

#Median regression for different distributions

modelN  = lqr(BD~FL,dist = "normal")
modelT  = lqr(BD~FL,dist = "t")
modelL  = lqr(BD~FL,dist = "laplace")
modelS  = lqr(BD~FL,dist = "slash")
modelC  = lqr(BD~FL,dist = "cont" )

#Comparing AIC criterias
modelN$AIC;modelT$AIC;modelL$AIC;modelS$AIC;modelC$AIC

# best model based on BIC
best.lqr(BD~FL,criterion = "BIC")

#Let's use a grid of quantiles for the Student's t distribution
modelfull = lqr(BD~FL,data = crabs,
                p = seq(from = 0.10,to = 0.90,by = 0.05),
                dist = "t") # silent = FALSE

#Plotting quantiles 0.10,0.25,0.50,0.75 and 0.90
if(TRUE){
  plot(FL,BD,xlab = "Frontal lobe size"
       ,ylab = "Body depth", main = "Quantile Regression",pch=16)
  colvec = c(2,2,3,3,4)
  imodel = c(1,17,4,14,9)
```

```
  for(i in 1:5){
    abline(a = modelfull[[imodel[i]]]$beta[1],
           b = modelfull[[imodel[i]]]$beta[2],
           lwd=2,col=colvec[i])
  }
  legend(x = "topleft",
         legend = rev(c("0.10","0.25","0.50","0.75","0.90")),
         lwd = 2,col = c(2,3,4,3,2))
}
```

`resistance` *Tumor-cell resistance to death*

Description

Artificial dataset. The experiment consists in measure the resistance to death of two types of tumor-cells over different doses of a experimental drug. The data was created considering a null intercept and a cubic polinomial for the dose.

Format

This data frame contains the following columns:

dose Quantity of dose of an experimental drug.

type Type of tumor-cell. Type A and B.

score Bounded response between 0 and 4.

Details

This dataset was generated in order to be fitted with a logistic quantile regression since the response is bounded.

`SKD` *Skew Family Distributions*

Description

Density, distribution function, quantile function and random generation for a Skew Family Distribution useful for quantile regression. This family of distribution includes skewed versions of the Normal, Student's t, Laplace, Slash and Contaminated Normal distribution, all with location parameter equal to `mu`, scale parameter `sigma` and skewness parameter `p`.

Usage

```
dSKD(y, mu = 0, sigma = 1, p = 0.5, dist = "normal", nu = "", gamma = "")
pSKD(q, mu = 0, sigma = 1, p = 0.5, dist = "normal", nu = "", gamma = "",
lower.tail = TRUE)
qSKD(prob, mu = 0, sigma = 1, p = 0.5, dist = "normal", nu = "", gamma = "",
lower.tail = TRUE)
rSKD(n, mu = 0, sigma = 1, p = 0.5, dist = "normal", nu = "", gamma = "")
```

Arguments

y, q	vector of quantiles.
prob	vector of probabilities.
n	number of observations.
mu	location parameter.
sigma	scale parameter.
p	skewness parameter.
dist	represents the distribution to be used for the error term. The values are normal for Normal distribution, t for Student's t distribution, laplace for Laplace distribution, slash for Slash distribution and cont for the Contaminated normal distribution.
nu	It represents the degrees of freedom when dist = t. For the Slash distribution (dist = slash) it is a shape parameter $\nu > 0$. For the Contaminated Normal distribution, ν is the parameter that represents the percentage of outliers. When is not provided, we use the MLE.
gamma	It represents a scale factor for the contaminated normal distribution. When is not provided, we use the MLE.
lower.tail	logical; if TRUE (default), probabilities are P[X $\leq$ x] otherwise, P[X > x].

Details

If mu, sigma, p or dist are not specified they assume the default values of 0, 1, 0.5 and normal, respectively, belonging to the Symmetric Standard Normal Distribution denoted by $SKN(0, 1, 0.5)$.

The scale parameter sigma must be positive and non zero. The skew parameter p must be between zero and one (0<p<1).

This family of distributions generalize the skew distributions in Wichitaksorn et.al. (2014) as an scale mixture of skew normal distribution. Also the Three-Parameter Asymmetric Laplace Distribution defined in Koenker and Machado (1999) is a special case.

Value

dSKD gives the density, pSKD gives the distribution function, qSKD gives the quantile function, and rSKD generates a random sample.

The length of the result is determined by n for rSKD, and is the maximum of the lengths of the numerical arguments for the other functions dSKD, pSKD and qSKD.

Note

The numerical arguments other than n are recycled to the length of the result.

Author(s)

Christian E. Galarza <<cgalarza88@gmail.com>>, Luis Benites <<lsanchez@ime.usp.br>> and Victor H. Lachos <<hlachos@ime.unicamp.br>>

Maintainer: Christian E. Galarza <<cgalarza88@gmail.com>>

References

Galarza, C., Lachos, V. H., Cabral, C. R. B., & Castro, C. L. (2017). Robust quantile regression using a generalized class of skewed distributions. Stat, 6(1), 113-130.

Wichitaksorn, N., Choy, S. T., & Gerlach, R. (2014). A generalized class of skew distributions and associated robust quantile regression models. Canadian Journal of Statistics, 42(4), 579-596.

See Also

lqr,ais

Examples

```
## Let's plot (Normal Vs. Student-t's with 4 df)
##Density
sseq = seq(15,65,length.out = 1000)
dens = dSKD(y=sseq,mu=50,sigma=3,p=0.75)
plot(sseq,dens,type="l",lwd=2,col="red",xlab="x",ylab="f(x)", main="Normal Vs. t(4) densities")
dens2 = dSKD(y=sseq,mu=50,sigma=3,p=0.75,dist="t",nu=4)
lines(sseq,dens2,type="l",lwd=2,col="blue",lty=2)

## Distribution Function
df = pSKD(q=sseq,mu=50,sigma=3,p=0.75,dist = "laplace")
plot(sseq,df,type="l",lwd=2,col="blue",xlab="x",ylab="F(x)", main="Laplace Distribution function")
abline(h=1,lty=2)

##Inverse Distribution Function
prob = seq(0.001,0.999,length.out = 1000)
idf = qSKD(prob=prob,mu=50,sigma=3,p=0.25,dist="cont",nu=0.3,gamma=0.1) # 1 min appox
plot(prob,idf,type="l",lwd=2,col="gray30",xlab="x",ylab=expression(F^{-1}~(x)))
title(main="Skew Cont. Normal Inverse Distribution function")
abline(v=c(0,1),lty=2)

#Random Sample Histogram
sample = rSKD(n=20000,mu=50,sigma=3,p=0.2,dist="slash",nu=3)
seqq2 = seq(25,100,length.out = 1000)
dens3 = dSKD(y=seqq2,mu=50,sigma=3,p=0.2,dist="slash",nu=3)
hist(sample,breaks = 70,freq = FALSE,ylim=c(0,1.05*max(dens3,na.rm = TRUE)),main="")
title(main="Histogram and True density")
lines(seqq2,dens3,col="blue",lwd=2)
```

Index

∗ **Distribution**
dist.Truncated, 8
∗ **bounded**
Log.best.lqr, 10
Log.lqr, 13
lqr-package, 2
∗ **datasets**
ais, 3
resistance, 23
∗ **distribution**
Egig, 9
SKD, 23
∗ **generalized gaussian**
Egig, 9
∗ **logistic**
Log.best.lqr, 10
Log.lqr, 13
lqr-package, 2
∗ **package**
best.lqr, 3
cens.lqr, 5
lqr, 18
lqr-package, 2
SKD, 23
∗ **quantile**
best.lqr, 3
cens.lqr, 5
Log.best.lqr, 10
Log.lqr, 13
lqr, 18
lqr-package, 2
SKD, 23
∗ **regression**
best.lqr, 3
cens.lqr, 5
Log.best.lqr, 10
Log.lqr, 13
lqr, 18
lqr-package, 2
∗ **skewed**
lqr-package, 2
∗ **skew**
best.lqr, 3
cens.lqr, 5
Log.best.lqr, 10
Log.lqr, 13
lqr, 18
SKD, 23

ais, *2*, 3, *25*

best.lqr, *2*, 3, *7*, *10*, *12*, *16*, *21*

cens.lqr, 5, *21*

dist.Truncated, 8
dSKD, *5*, *7*, *12*, *16*, *21*
dSKD *(*SKD*)*, 23
dtrunc *(*dist.Truncated*)*, 8

Egig, 9
extrunc *(*dist.Truncated*)*, 8

grad, *9*

Log.best.lqr, *2*, *5*, *7*, 10, *16*, *21*
Log.lqr, *2*, *5*, *7*, *12*, 13, *21*
lqr, *2*, *5*, *7*, *9*, 18, *25*
lqr-package, 2

pSKD *(*SKD*)*, 23
ptrunc *(*dist.Truncated*)*, 8

qSKD *(*SKD*)*, 23
qtrunc *(*dist.Truncated*)*, 8

resistance, 23
rSKD *(*SKD*)*, 23
rtrunc *(*dist.Truncated*)*, 8

SKD, *2*, *9*, 23

vartrunc *(*dist.Truncated*)*, 8