Package ‘lqr’
February 1, 2018

Type Package
Title Robust Linear Quantile Regression
Version 1.7
Date 2018-01-31
Author Christian E. Galarza, Luis Benites, Victor H. Lachos
Maintainer Christian E. Galarza <cgalarza88@gmail.com>
Imports graphics, stats, ghyp, spatstat
Suggests ald, qrLMM, qrNLMM
Description It fits a robust linear quantile regression model using a new
family of zero-quantile distributions for the error term as in Galarza et.al.(2017) <doi:10.1002/sta4.140>. 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
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 2018-02-01 16:13:20 UTC

R topics documented:

lqr-package .................................................. 2
ais ............................................................. 3
best.lqr ......................................................... 3
dist.Truncated ................................................ 5
Log.best.lqr ................................................... 7
Log.lqr ......................................................... 10
lqr .............................................................. 15
resistance ...................................................... 19
SKD ............................................................ 20
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

Package: lqr
Type: Package
Version: 1.0
Date: 2015-10-29
License: GPL (>=2)

~ An overview of how to use the package, including the most important functions ~

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


See Also

SKD, Log.best.lqr, Log.lqr.best.lqr, lqr, ais, QRLMM, QRNLMM
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
- SSSF sum of skin folds
- Bfat Percent body fat
- Label Case Labels
- Sport Sport

References


---

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(y, x, p = 0.5, precision = 10^-6, criterion = "AIC")

Arguments

y
the response vector of dimension n where n is the total of observations.

x
design matrix for the fixed effects of dimension Nxd where d represents the number of fixed effects including the intercept, if considered.

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>>
dist.Truncated

References


See Also

lqr, Log.lqr, Log.best.lqr, dSKD, QRLLM, QRNLMM

Examples

```r
## Not run:
data(crabs, package = "MASS")

crabs$sex <- as.character(crabs$sex)
crabs$sex[crabs$sex=="M"] = 0
crabs$sex[crabs$sex=="F"] = 1

crabs$sex = as.numeric(crabs$sex)

attach(crabs)

## Setting
y <- BD
x <- cbind(1, FL, sex)

# Finding the best model for the 3rd quartile based on Akaike criterion
res = best.lqr(y, x, p = 0.75, criterion = "AIC")

## End(Not run)
```

---

dist.Truncated  Truncated Distributions

Description

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

Usage

```r
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 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


See Also

- lqr, SKD.

Examples

```r
x <- seq(-0.5, 0.5, by = 0.1)
y <- dtrunc(x, "norm", a=-0.5, b=0.5, mean=0, sd=2)
```
**Log.best.lqr**

**Best Fit in Robust Logistic Linear Quantile Regression**

**Description**

It performs the logistic transformation in Bottai et.al. (2009) (see references) for estimating quantiles for a bounded response. Once the response is transformed, it uses the `best.lqr` function.

**Usage**

```r
Log.best.lqr(y, x, p=0.5, a=0, b=1, epsilon = 0.001, precision = 10^-6, criterion = "AIC")
```

**Arguments**

- `y`: 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 $y$
- `y`: the response vector of dimension $n$ where $n$ is the total of observations.
- `x`: design matrix for the fixed effects of dimension $N \times d$ where $d$ represents the number of fixed effects including the intercept, if considered.
- `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 chosen 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

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

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 $\beta_p$ 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 continuous, the respective $\beta$ coefficient can be interpreted 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 <<l.sanchez@ime.usp.br>> and Victor H. Lachos <<h.lachos@ime.unicamp.br>>

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

**References**


See Also

Log.lqr, best.lqr, dSKD, QRLMM, QRNLMM

Examples

```r
## Not run:
# Load the data
data(resistance)
attach(resistance)

# EXAMPLE 1.1

# Comparing the resistance 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(y = score, x = cbind(1,type), 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) # better fitted

# EXAMPLE 1.2

# Comparing the resistance 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
xx1 = dose
xx2 = dose^2
xx3 = dose^3
res31 = best.lqr(y = score, x = cbind(1,xx1,xx2,xx3), p = 0.05)
res32 = best.lqr(y = score, x = cbind(1,xx1,xx2,xx3), p = 0.50)
res33 = best.lqr(y = score, x = cbind(1,xx1,xx2,xx3), p = 0.95)
res3 = list(res31, res32, res33)
```
seqq = seq(min(dose), max(dose), length.out = 1000)
d = matrix(data = NA, nrow = 1000, ncol = 3)
for (i in 1:3)
{
    d[i,] = rep(res3[[i]]$beta[1], 1000) + res3[[i]]$beta[2]*seqq +
    res3[[i]]$beta[3]*seqq^2 + res3[[i]]$beta[4]*seqq^3
}

plot(dose, score, ylim = c(-1, 5), col = "gray"); abline(h = c(0, 4), lty = 2)
lines(seqq, d[, 1], lwd = 1, col = 2)
lines(seqq, d[, 2], lwd = 1, col = 1)
lines(seqq, d[, 3], lwd = 1, col = 2)

# Using logistic quantile regression for obtaining predictions inside bounds

res41 = Log.best.lqr(y = score, x = cbind(x1, xx1, xx2, xx3), a = 0, b = 4, p = 0.05)
res42 = Log.best.lqr(y = score, x = cbind(x1, xx1, xx2, xx3), a = 0, b = 4, p = 0.50)
res43 = Log.best.lqr(y = score, x = cbind(x1, xx1, xx2, xx3), a = 0, b = 4, p = 0.95)
res4 = list(res41, res42, res43)
d = matrix(data = NA, nrow = 1000, ncol = 3)
for (i in 1:3)
{
    d[i,] = rep(res4[[i]]$beta[1], 1000) + res4[[i]]$beta[2]*seqq +
    res4[[i]]$beta[3]*seqq^2 + res4[[i]]$beta[4]*seqq^3
}

# Computing quantiles for the original response (Inverse transformation)
pred = function(predlog, a, b)
{
    return((b*exp(predlog)+a)/(1+exp(predlog)))
}
for (i in 1:3)
{
    d[i,] = pred(d[i,], a = 0, b = 4)
}

# No more prediction curves out of bounds
plot(dose, score, ylim = c(0, 4), col = "gray"); abline(h = c(0, 4), lty = 2)
lines(seqq, d[, 1], lwd = 1, col = 2)
lines(seqq, d[, 2], lwd = 1, col = 1)
lines(seqq, d[, 3], lwd = 1, col = 2)

## End (Not run)
**Description**

It performs the logistic transformation in Bottai et.al. (2009) (see references) for estimating quantiles for a bounded response. Once the response is transformed, it uses the lqr function.

**Usage**

\[
\text{Log.lqr}(y, x, p=0.5, a=0, b=1, \text{dist} = \text{"normal"}, \nu = \text{""}, \text{gama} = \text{""}, \text{precision} = 10^{-6}, \\
\text{epsilon} = 0.001, \text{CI} = 0.95)
\]

**Arguments**

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

- lower bound for the response (default = 0)
- upper bound for the response (default = 1)
- a small quantity \(\epsilon > 0\) that ensures that the logistic transform is defined for all values of \(y\)
- the response vector of dimension \(n\) where \(n\) is the total of observations.
- design matrix for the fixed effects of dimension \(N \times d\) where \(d\) represents the number of fixed effects including the intercept, if considered.
- An unique quantile or a set of quantiles related to the quantile regression.
- 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.
- It represents the degrees of freedom when \(\text{dist} = t\). For the Slash distribution (\(\text{dist} = \text{slash}\)) it is a shape parameter \(\nu > 0\). For the Contaminated Normal distribution, \(\nu\) is the parameter that represents the percentage of outliers. When is not provided, we use the MLE.
- It represents a scale factor for the contaminated normal distribution. When is not provided, we use the MLE.
- The convergence maximum error permitted. By default is \(10^{-6}\).
- Confidence to be used for the Confidence Interval when a grid of quantiles is provided. Default = 0.95.

**Details**

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

\[
h(y) = \text{logit}(y) = \log\left(\frac{y - a}{b - y}\right)
\]

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 $\beta_p$ 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 continuous, the respective $\beta$ coefficient can be interpreted 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**


See Also

Log.best.lqr, best.lqr, dSKD, QRLMM, QRNLMM

Examples

### Not run:
```r
# Load the data
data(resistance)
attach(resistance)

# EXAMPLE 1.1

# Comparing the resistance to death of two types of tumor-cells.
# The response is a score in [0,4].
boxplot(score~type, ylab="score", xlab="type")

# Student's median logistic quantile regression
res = Log.lqr(y = score, x = cbind(1, type), 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 resistance 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)

# Slash (Non logistic) quantile regression for quantiles 0.05, 0.50 and 0.95
xx1 = dose
xx2 = dose^2
xx3 = dose^3
res3 = lqr(y = score, x = cbind(1, xx1, xx2, xx3), p = c(0.05, 0.50, 0.95), dist="slash")
seqq = seq(min(dose), max(dose), length.out = 1000)
dd = matrix(data = NA, nrow = 1000, ncol = 3)
for(i in 1:3)
```

Log.lqr
\{ 
  dd[,i] = rep(res3[[i]]$beta[1],1000) + res3[[i]]$beta[2]*seqq + 
           res3[[i]]$beta[3]*seqq^2 + res3[[i]]$beta[4]*seqq^3  
\}

plot(dose,score,ylim=c(-1,5),col="gray");abline(h=c(0,4),lty=2)
lines(seqq,dd[,1],lwd=1,col=2)
lines(seqq,dd[,2],lwd=1,col=1)
lines(seqq,dd[,3],lwd=1,col=2)

#Using logistic quantile regression for obtaining predictions inside bounds

res4 = Log.lqr(y = score,x = cbind(1, xx1, xx2, xx3), a = 0, b = 4, p = c(0.05, 0.50, 0.95), dist="slash")

for(i in 1:3)
  { 
    dd[,i] = rep(res4[[i]]$beta[1],1000) + res4[[i]]$beta[2]*seqq + 
            res4[[i]]$beta[3]*seqq^2 + res4[[i]]$beta[4]*seqq^3  
  }

#Computing quantiles for the original response (Inverse transformation)

pred = function(predlog,a,b)
  { 
    return((b*exp(predlog)+a)/(1+exp(predlog)))  
  }

for(i in 1:3)
  { 
    dd[,i] = pred(dd[,i],a=0,b=4)  
  }

#No more prediction curves out of bounds

plot(dose,score,ylim=c(0,4),col="gray");abline(h=c(0,4),lty=2)
lines(seqq,dd[,1],lwd=1,col=2)
lines(seqq,dd[,2],lwd=1,col=1)
lines(seqq,dd[,3],lwd=1,col=2)

#EXAMPLE 1.3

###############

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

typeB = 1*(type=="B")
res5 = Log.lqr(y = score,x = cbind(1, xx1, xx2, xx3, typeB, typeB*xx1), a = 0, b = 4, 
        p = seq(from = 0.05, to = 0.95, by = 0.05), dist = "t")

dda = ddB = matrix(data = NA,nrow = 1000,ncol = 5)
for(i in 1:5)
  { 
    k = c(2,5,10,15,18)[i]
    dda[,i] = rep(res5[[k]]$beta[1],1000) + res5[[k]]$beta[2]*seqq + res5[[k]]$beta[3]* 
              seqq^2 + res5[[k]]$beta[4]*seqq^3
    ddB[,i] = rep(res5[[k]]$beta[2],1000) + (res5[[k]]$beta[2] + res5[[k]]$beta[6])*
```r
seqq + res5[[k]]*beta[3]*seqq^2 + res5[[k]]*beta[4]*seqq^3 + res5[[k]]*beta[5]
```

# Computing quantiles for the original response (Inverse transformation)

```r
for(i in 1:5)
{
  ddA[i] = pred(ddA[i], a=0, b=4)
  ddB[i] = pred(ddB[i], a=0, b=4)
}
```

# Such a beautiful plot
par(mfrow=c(1,2))
plot(dose, score, ylim=c(0,4), col=c((type == "B")*8+(type == "A")*1), main="Type A")
abline(h=c(0,4), lty=2)
lines(seqq, ddA[,1], lwd=2, col=2)
lines(seqq, ddA[,2], lwd=1, col=4)
lines(seqq, ddA[,3], lwd=2, col=1)
lines(seqq, ddA[,4], lwd=1, col=4)
lines(seqq, ddA[,5], lwd=2, col=2)
legend(x = 0, y = 4, legend = c("p=0.10", "p=0.25", "p=0.50", "p=0.75", "p=0.90"),
  col = c(2,4,1,4,2), lwd = c(2,1,2,1,2), bty = "n", cex = 0.65)
```

## 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**

```r
lqr(y,x,p=0.5,dist="normal",nu="",gama="",precision=10^-6,envelope=FALSE,CI=0.95)
```

```r
#lqr(y, x, p = 0.5, dist = "normal")
```
Arguments

\( y \)
the response vector of dimension \( n \) where \( n \) is the total of observations.

\( x \)
design matrix for the fixed effects of dimension \( N \times d \) where \( d \) represents the number of fixed effects including the intercept, if considered.

\( p \)
An unique quantile or a set of quantiles related to the quantile regression.

\( \text{dist} \)
represents the distribution to be used for the error term. The values are \text{normal} for Normal distribution, \text{t} for Student’s t distribution, \text{laplace} for Laplace distribution, \text{slash} for Slash distribution and \text{cont} for the Contaminated normal distribution.

\( \nu \)
It represents the degrees of freedom when \( \text{dist} = \text{t} \). For the Slash distribution (\( \text{dist} = \text{slash} \)) it is a shape parameter \( \nu > 0 \). For the Contaminated Normal distribution, \( \nu \) 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.

\( \text{precision} \)
The convergence maximum error permitted. By default is \( 10^{-6} \).

\( \text{envelope} \)
if \( \text{TRUE} \), it will show a confidence envelope for a curve based on bootstrap replicates. By default it is \( \text{TRUE} \) when just one quantile is provided. If a grid of quantiles is provided it will be \( \text{FALSE} \) by default.

\( \text{CI} \)
Confidence to be used for the Confidence Interval when a grid of quantiles is provided. Default = 0.95.

Details

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

Value

\( \text{iter} \)
number of iterations.

\( \text{criteria} \)
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.

\( \text{SE} \)
Standard Error estimates.

\( \text{table} \)
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

See Also
best.lqr,Log.lqr,Log.best.lqr,dSKD,QRLMM,QRNLMM

Examples
```r
## Not run:
#Example 1
#Load the data
data(ais)
attach(ais)

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

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

modelF  = lqr(y[Sex==1], x[Sex==1,1:2])
modelM  = lqr(y[Sex==0], x[Sex==0,1:2])
plot(LBM,BMI,col=Sex+1,ylab="Lean Body Mass",xlab="Body4 Mass Index",main="Quantile Regression")
abline(a = modelF$beta[1],b = modelF$beta[2],lwd=2,col=2)
abline(a = modelM$beta[1],b = modelM$beta[2],lwd=2,col=4)
```
Comparing some models for median regression

```r
x <- cbind(1, LBM)

modelN = lqr(y, x, dist = "normal")
modelT = lqr(y, x, dist = "t")
modelL = lqr(y, x, dist = "laplace")

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

# This could be automatically done using best.lqr()
best.model = best.lqr(y, x, p = 0.75, criterion = "AIC")

# Let's use a grid of quantiles
modelfull = lqr(y, x, p = seq(from = 0.10, to = 0.90, by = 0.05), dist = "normal")

# Plotting quantiles 0.10, 0.25, 0.50, 0.75 and 0.90
plot(LBM, BMI, xlab = "Lean Body Mass", ylab = "Body Mass Index", main = "Quantile Regression", pch = 16)
abline(a = modelfull[[1]]$beta[1], b = modelfull[[1]]$beta[2], lwd = 2, lty = 1, col = 2)
abline(a = modelfull[[17]]$beta[1], b = modelfull[[17]]$beta[2], lwd = 2, lty = 1, col = 2)
abline(a = modelfull[[4]]$beta[1], b = modelfull[[4]]$beta[2], lwd = 2, lty = 1, col = 3)
abline(a = modelfull[[14]]$beta[1], b = modelfull[[14]]$beta[2], lwd = 2, lty = 1, col = 3)
abline(a = modelfull[[9]]$beta[1], b = modelfull[[9]]$beta[2], lwd = 2, lty = 1, col = 4)

# Example 2
# Load the data
data(crabs, package = "MASS")

crabs$sex <- as.character(crabs$sex)
crabs$sex[crabs$sex == "M"] = 0
crabs$sex[crabs$sex == "F"] = 1

attach(crabs)
head(crabs)

# Setting
y <- BD
x <- cbind(1, FL, sex)
plot(crabs)

# Fitting a median regression with Normal errors (by default)
modelF = lqr(y[sex == 1], x[sex == 1, 2])
modelM = lqr(y[sex == 0], x[sex == 0, 1:2])
plot(FL, BD, col = sex + 1, xlab = "Frontal lobe size", ylab = "Body depth", main = "Quantile Regression")
abline(a = modelF$beta[1], b = modelF$beta[2], lwd = 2, col = 2)
abline(a = modelM$beta[1], b = modelM$beta[2], lwd = 2, col = 4)
```
resistance

#COMPARING SOME MODELS for median regression
x <- cbind(1,FL)

modelN = lqr(y,x,dist = "normal")
modelT = lqr(y,x,dist = "t")
modelL = lqr(y,x,dist = "laplace")
modelS = lqr(y,x,dist = "slash")
modelC = lqr(y,x,dist = "cont" )

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

#Let's use a grid of quantiles
modelfull = lqr(y,x,p = seq(from = 0.10,to = 0.90,by = 0.05),dist = "t")

#Plotting quantiles 0.10,0.25,0.50,0.75 and 0.90
plot(FL,BO,xlab = "Frontal lobe size", ylab = "Body depth", main = "Quantile Regression",pch=16)
abline(a = modelfull[[1]]$beta[1],b = modelfull[[1]]$beta[2],lwd=2,lty=1,col=2)
abline(a = modelfull[[17]]$beta[1],b = modelfull[[17]]$beta[2],lwd=2,lty=1,col=2)
abline(a = modelfull[[4]]$beta[1],b = modelfull[[4]]$beta[2],lwd=2,lty=1,col=3)
abline(a = modelfull[[14]]$beta[1],b = modelfull[[14]]$beta[2],lwd=2,lty=1,col=3)
abline(a = modelfull[[9]]$beta[1],b = modelfull[[9]]$beta[2],lwd=2,lty=1,col=4)

## End(Not run)

resistance Tumor-cell resistance to death

Description

The experiment consists in measure the resistance to death of two types of tumor-cells over different doses of a experimental drug. P.D.: Simulated dataset.

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.
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

\[
\begin{align*}
dSKD(y, \mu = 0, \sigma = 1, p = 0.5, \text{dist} = "\text{normal}" , \nu = ", gama = " , \text{lower.tail} = \text{TRUE}) \\
pSKD(q, \mu = 0, \sigma = 1, p = 0.5, \text{dist} = "\text{normal}" , \nu = ", gama = " , \text{lower.tail} = \text{TRUE}) \\
qSKD(prob, \mu = 0, \sigma = 1, p = 0.5, \text{dist} = "\text{normal}" , \nu = ", gama = " , \text{lower.tail} = \text{TRUE}) \\
rSKD(n, \mu = 0, \sigma = 1, p = 0.5, \text{dist} = "\text{normal}" , \nu = ", gama = " \\
\end{align*}
\]

Arguments

- \( y, q \) vector of quantiles.
- \( \text{prob} \) vector of probabilities.
- \( n \) number of observations.
- \( \mu \) location parameter.
- \( \sigma \) scale parameter.
- \( p \) skewness parameter.
- \( \text{dist} \) 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 \( \text{dist} = \text{t} \). For the Slash distribution (\( \text{dist} = \text{slash} \)) it is a shape parameter \( \nu > 0 \). For the Contaminated Normal distribution, \( \nu \) is the parameter that represents the percentage of outliers. When is not provided, we use the MLE.
- \( gama \) It represents a scale factor for the contaminated normal distribution. When is not provided, we use the MLE.
- \( \text{lower.tail} \) logical; if \( \text{TRUE} \) (default), probabilities are \( P[X \leq x] \) otherwise, \( P[X > x] \).

Details

If \( \mu, \sigma, p \) or \( \text{dist} \) 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 generalizes 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**


**See Also**

lqr, ais

**Examples**

```R
# Not run:
# 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)
```
```r
## 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, gama=0.1)  # 1 min approx
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)

## End(Not run)
```
Index

*Topic Distribution
  dist.Truncated, 5
*Topic bounded
  Log.best.lqr, 7
  Log.lqr, 10
  lqr-package, 2
*Topic datasets
  ais, 3
  resistance, 19
*Topic distribution
  SKD, 20
*Topic logistic
  Log.best.lqr, 7
  Log.lqr, 10
  lqr-package, 2
*Topic package
  best.lqr, 3
  lqr, 15
  lqr-package, 2
  SKD, 20
*Topic quantile
  best.lqr, 3
  Log.best.lqr, 7
  Log.lqr, 10
  lqr, 15
  lqr-package, 2
  SKD, 20
*Topic regression
  best.lqr, 3
  Log.best.lqr, 7
  Log.lqr, 10
  lqr, 15
  lqr-package, 2
*Topic skewed
  lqr-package, 2
*Topic skew
  best.lqr, 3
  Log.best.lqr, 7
  Log.lqr, 10
  lqr, 15
  SKD, 20
  rskd (SKD), 20
  rtrunc (dist.Truncated), 5
  resistance, 19
  vartrunc (dist.Truncated), 5
lqr, 15
SKD, 20
ais, 2, 3, 21
best.lqr, 2, 3, 9, 13, 17
dist.Truncated, 5
dSKD, 5, 9, 13, 17
dSKD (SKD), 20
dtrunc (dist.Truncated), 5
eextrunc (dist.Truncated), 5
Log.best.lqr, 2, 5, 7, 13, 17
Log.lqr, 2, 5, 9, 10, 17
lqr, 2, 5, 6, 15, 21
lqr-package, 2
pSKD (SKD), 20
ptrunc (dist.Truncated), 5
QRLMM, 2, 5, 9, 13, 17
QRNLMM, 2, 5, 9, 13, 17
qSKD (SKD), 20
qtrunc (dist.Truncated), 5
resistance, 19
rSKD (SKD), 20
rtrunc (dist.Truncated), 5
SKD, 2, 6, 20
vartrunc (dist.Truncated), 5