Package ‘NormalLaplace’
November 26, 2018

Version 0.3-0
Date 2018-11-26
Title The Normal Laplace Distribution
Author David Scott <d.scott@auckland.ac.nz>, Jason Shicong Fu and
    Simon Potter
Maintainer David Scott <d.scott@auckland.ac.nz>
Depends R (>= 3.0.1),
Imports DistributionUtils, GeneralizedHyperbolic
Suggests RUnit
Encoding latin1
Description Functions for the normal Laplace distribution. The package is
    under development and provides only limited functionality.
    Density, distribution and quantile functions, random number generation,
    and moments are provided.
License GPL (>= 2)
Repository CRAN
Repository/R-Forge/Project rmetrics
Repository/R-Forge/Revision 4777
Date/Publication 2018-11-26 12:00:03 UTC
NeedsCompilation no

R topics documented:

  NormalLaplace-package ........................................ 2
  MillsRatio .................................................... 2
  nlCheckPars .................................................. 3
  nlFit ........................................................ 4
  nlFitStart .................................................... 7
  nlPlots ....................................................... 9
  NormalLaplaceDistribution .................................... 10
  NormalLaplaceMeanVar ........................................ 12
  summary.nlFit ............................................... 14
Description

This package provides a collection of functions for Normal Laplace distributions. Functions are provided for the density function, distribution function, quantiles and random number generation. The mean, variance, skewness and kurtosis of a given Normal Laplace distribution are given by \texttt{nlmean}, \texttt{nlVar}, \texttt{nlSkew}, and \texttt{nlKurt} respectively.

Author(s)

David Scott <d.scott@auckland.ac.nz>, Jason Shicong Fu
Maintainer: David Scott <d.scott@auckland.ac.nz>

References


See Also

dnl, millsR, NormalLaplaceMeanVar

Description

Calculates the Mills ratio

Usage

\texttt{millsR(y, log = FALSE)}

Arguments

\begin{itemize}
  \item \texttt{y} \hspace{1cm} \text{Numeric. Value at which the Mills’ Ratio is evaluated.}
  \item \texttt{log} \hspace{1cm} \text{Logical. If log = TRUE, Mills’ Ratios are given as log(millsR).}
\end{itemize}

Details

The function calculates the Mills’ Ratio. Since the Mill’s Ratio converges to zero for large positive \( z \) and infinity for large negative \( z \). The range over which the logarithm of the Mill’s ratio may be calculated is greater than that for which the Mill’s ratio itself may be calculated.
The Mills’ Ratio is
\[ R(z) = \frac{1 - \Phi(z)}{\phi(z)} \]
where \( \Phi(z) \) and \( \phi(z) \) are respectively the distribution function and density function of the standard normal distribution.

Author(s)
David Scott <d.scott@auckland.ac.nz>, Jason Shicong Fu

Examples

```r
## compare millsR calculated directly with the millsR calculated
## by transforming to log scale and then back-transformed
millsR(1:10)
exp(millsR(1:10, log = TRUE))
exp(millsR(10*(1:10)))
exp(millsR(10*(1:10), log = TRUE))
```

nlCheckPars

Check Parameters of the Normal Laplace Distribution

Description
Given a set of parameters for the normal Laplace distribution, the function checks the validity of each parameter and if they and if they correspond to the boundary cases.

Usage

```
nlCheckPars(param)
```

Arguments

- `param` Numeric. Parameter values for the normal Laplace distribution.

Details
The vector `param` takes the form `c(mu, sigma, alpha, beta)`.
If any of `sigma`, `alpha` or `beta` is negative or NA, an error is returned.

Author(s)
David Scott <d.scott@auckland.ac.nz>, Simon Potter
References


Examples

```r
## Correct parameters
nlCheckPars(c(0, 1.5, 1, 2))
nlCheckPars(c(3, 1, 1.5, 2))

## Incorrect parameters, each error providing a different error message
nlCheckPars(c(2, -1, 1, 1))  # invalid sigma
nlCheckPars(c(2, 1, -1, 2))  # invalid alpha
nlCheckPars(c(0, 1, 2, -1))  # invalid beta
nlCheckPars(c(0, -0.01, -0.1, 1))  # sigma and alpha incorrect
nlCheckPars(c(2, -0.5, 1, -0.2))  # sigma and beta incorrect
nlCheckPars(c(1, 1, -0.2, -1))  # alpha and beta incorrect
nlCheckPars(c(0, -0.1, -0.2, -0.3))  # all three parameters erroneous
nlCheckPars(c(0.5, NA, 1, 1))  # NA introduced
nlCheckPars(c(-1, 1, 1))  # incorrect number of parameters
```

---

**nlFit**  
*Fit the Normal Laplace Distribution to Data*

**Description**

Fits a normal Laplace distribution to data. Displays the histogram, log-histogram (both with fitted densities), Q-Q plot and P-P plot for the fit which has the maximum likelihood.

**Usage**

```r
nlFit(x, freq = NULL, breaks = "FD", paramStart = NULL,
      startMethod = "Nelder-Mead",
      startValues = c("MoM", "US"),
      method = c("Nelder-Mead", "BFGS", "L-BFGS-B",
                 "nlm", "nlminb"),
      hessian = FALSE,
      plots = FALSE, printOut = FALSE,
      controlBFGS = list(maxit = 200),
      controlBFGSB = list(maxit = 200),
      controlNLINB = list(),
      controlNM = list(maxit = 1000),
      maxitNM = 1500, ...)
```

## S3 method for class 'nlFit'

```r
print(x, digits = max(3, getOption("digits") - 3), ...)
```

## S3 method for class 'nlFit'

```r
plot(x, which = 1:4,
     plotTitles = paste(c("Histogram of ", "Log-Histogram of ",
```

```
"Q-Q Plot of ","P-P Plot of ", x$obsName,
   sep = " ",
   ask = prod(par("mfcol")) < length(which) & dev.interactive(), ...)
## S3 method for class 'nlFit'
coef(object, ...)
## S3 method for class 'nlFit'
vcov(object, ...)

Arguments

x          Data vector for nlFit.
freq       A vector of weights with length equal to length(x).
breaks      Breaks for plotted histogram, defaults to those generated by hist(x, right = FALSE, plot = FALSE).
paramStart  A user specified starting parameter vector taking the form c(mu, sigma, alpha, beta).
startMethod Method used by nlFitStart in calls to optim.
startValues Code giving the method of determining starting values for finding the maximum likelihood estimate of the parameters.
method      Different optimisation methods to consider. See Details.
hessian     Logical. If TRUE the value of the Hessian is returned.
plots       Logical. If FALSE the printing of the histogram, log-histogram, Q-Q plot and P-P plot are suppressed.
printOut    Logical. If FALSE the printing of the results of fitting will be suppressed.
controlBFGS A list of control parameters for optim when using the "BFGS" method of optimisation.
controlLBFGSB A list of control parameters for optim when using the "L-BFGS-B" method of optimisation.
controlNLMINB A list of control parameters for optim when using the "nlminb" method of optimisation.
controlNM   A list of control parameters for optim when using the "Nelder-Mead" method of optimisation.
maxitNLM    A positive integer specifying the maximum number of iterations that are to be undertaken when using the "nlm" method of optimisation.
object      Object of class "nlFit" for print.nlFit, plot.nlFit, coef.nlFit and vcov.nlFit.
digits      Desired number of digits to be shown when the object is printed.
which        If a subset of the plots if required, specify a subset of the numbers 1:4.
plotTitles  Titles to appear as the main title above the plots.
ask          Logical. If TRUE, the user is asked before each plot, see par(ask = ).
...          Passes arguments to par, hist, logHist, qqnl and ppnl. For the print, coef and vcov methods this parameter has no effect.
Details

startMethod must be "Nelder-Mead".

startValues can only be "MoM" when using the Method of Moments for estimation, or "US" for user-supplied parameters. For details regarding the use of paramStart, startMethod and startValues, see nlFitStart.

Three optimisation methods are available for use:

- "BFGS" Uses the quasi-Newton method "BFGS" as documented in optim.
- "L-BFGS-B" Uses the constrained method "L-BFGS-B" as documented in optim.
- "Nelder-Mead" Uses an implementation of the Nelder and Mead method as documented in optim.
- "nlm" Uses the nlm function in R.
- "nlminb" Uses the nlminb function in R, with constrained parameters.

For details on how to pass control information for optimisation using optim and nlm, see optim and nlm.

When method = "nlm" or method = "nlm" is used, warnings may be produced. However, these do not appear to be problematic.

Value

A list with components:

- param A vector giving the maximum likelihood estimate of parameters, as c(mu, sigma, alpha, beta).
- maxLik The value of maximised log-likelihood.
- hessian If hessian was set to TRUE, the value of the Hessian. Not present otherwise.
- method Optimisation method used.
- conv Convergence code. See the relevant documentation (either optim or nlm) for details on convergence.
- iter Number of iterations made by the optimisation routine.
- obs The data used to fit the normal Laplace distribution.
- obsName A character vector with the actual x argument name.
- paramStart Starting value of parameters returned by call to nlFitStart.
- svName Descriptive name for the method of finding start values.
- startValues Acronym for the method of finding start values.
- breaks The cell boundaries found by a call to hist.
- midpoints The cell midpoints found by a call to hist.
- empDens The estimated density found by a call to hist.

Author(s)

David Scott <d.scott@auckland.ac.nz>, Simon Potter
nlFitStart

See Also

optim, nlm, par, hist, logHist, qqnl, ppnl, dnl and nlFitStart.

Examples

```r
param <- c(0, 2, 1, 1)
dataVector <- rnl(1000, param = param)

## Let's see how well nlFit works
nlFit(dataVector)
nlFit(dataVector, plots = TRUE)
fit <- nlFit(dataVector)
par(mfrow = c(1, 2))
plot(fit, which = c(1, 3))  # See only histogram and Q-Q plot
```

nlFitStart

Find Starting Values for Fitting a Normal Laplace Distribution

Description

Finds starting values for input to a maximum likelihood routine for fitting a normal Laplace distribution to data.

Usage

```r
nlFitStart(x, breaks = "FD",
          paramStart = NULL,
          startValues = c("MoM", "US"),
          startMethodMoM = "Nelder-Mead", ...)
```

Arguments

- **x**: Data vector.
- **breaks**: Breaks for histogram. If missing, defaults to those generated by `hist(x, right = FALSE, plot = FALSE)`.
- **paramStart**: Starting values for parameter vector if `startValues = "US"`.
- **startValues**: Vector of the different starting value methods to consider. See Details.
- **startMethodMoM**: Method used by call to `optim` in finding method of moments estimates.
- **...**: Passes arguments to `optim`.

Description

Finds starting values for input to a maximum likelihood routine for fitting a normal Laplace distribution to data.

Usage

```r
nlFitStart(x, breaks = "FD",
          paramStart = NULL,
          startValues = c("MoM", "US"),
          startMethodMoM = "Nelder-Mead", ...)
```

Arguments

- **x**: Data vector.
- **breaks**: Breaks for histogram. If missing, defaults to those generated by `hist(x, right = FALSE, plot = FALSE)`.
- **paramStart**: Starting values for parameter vector if `startValues = "US"`.
- **startValues**: Vector of the different starting value methods to consider. See Details.
- **startMethodMoM**: Method used by call to `optim` in finding method of moments estimates.
- **...**: Passes arguments to `optim`.
Details

Possible values of the argument startValues are the following:

- "US" User-supplied.
- "MoM" Method of moments.

If startValues = "US" then a value must be supplied for paramStart.
If startValues = "MoM", nlFitStartMoM is called.
If startValues = "MoM" an initial optimisation is needed to find the starting values. These optimisations call optim.

Value

nlFitStart returns a list with components:

paramStart A vector with elements mu, sigma, alpha and beta giving the starting value of param.
xName A character string with the actual x argument name.
brakes The cell boundaries found by a call to hist.
midpoints The cell midpoints found by a call to hist.
empDens The estimated density found by a call to hist.

nlFitStartMoM returns only the method of moments estimates as a vector with elements mu, sigma, alpha and beta.

Author(s)

David Scott <d.scott@auckland.ac.nz>, Simon Potter

See Also

dnl, nlFit, hist, and optim.

Examples

param <- c(2, 2, 1, 1)
dataVector <- rnl(500, param = param)
nlFitStart(dataVector, startValues = "MoM")
**Description**

`qqnl` produces a normal Laplace Q-Q plot of the values in `y`.

`ppnl` produces a normal Laplace P-P (percent-percent) or probability plot of the values in `y`.

Graphical parameters may be given as arguments to `qqnl`, and `ppnl`.

**Usage**

```r
qqnl(y, mu = 0, sigma = 1, alpha = 1, beta = 1, 
    param = c(mu, sigma, alpha, beta), 
    main = "Normal Laplace Q-Q Plot", 
    xlab = "Theoretical Quantiles", 
    ylab = "Sample Quantiles", 
    plot.it = TRUE, line = TRUE, ...)
ppnl(y, mu = 0, sigma = 1, alpha = 1, beta = 1, 
    param = c(mu, sigma, alpha, beta), 
    main = "Normal Laplace P-P Plot", 
    xlab = "Uniform Quantiles", 
    ylab = "Probability-integral-transformed Data", 
    plot.it = TRUE, line = TRUE, ...)
```

**Arguments**

- `y` The data sample.
- `mu` \(\mu\) is the location parameter. By default this is set to 0.
- `sigma` \(\sigma\) is the variance parameter of the distribution. A default value of 1 has been set.
- `alpha` \(\alpha\) is a skewness parameter, with a default value of 1.
- `beta` \(\beta\) is a shape parameter, by default this is 1.
- `param` Parameters of the normal Laplace distribution.
- `xlab`, `ylab`, `main` Plot labels.
- `plot.it` Logical. Should the result be plotted?
- `line` Add line through origin with unit slope.
- `...` Further graphical parameters.

**Value**

For `qqnl` and `ppnl`, a list with components:

- `x` The x coordinates of the points that are to be plotted.
- `y` The y coordinates of the points that are to be plotted.
References


See Also

ppoints, dnl, nlfit

Examples

```r
par(mfrow = c(1, 2))
param <- c(2, 2, 1, 1)
y <- rnl(200, param) = param)
qqnl(y, param = param, line = FALSE)
abline(0, 1, col = 2)
ppnl(y, param = param)
```

NormalLaplaceDistribution

Normal Laplace Distribution

Description

Density function, distribution function, quantiles and random number generation for the normal Laplace distribution, with parameters $\mu$ (location), $\delta$ (scale), $\beta$ (skewness), and $\nu$ (shape).

Usage

```r
dnl(x, mu = 0, sigma = 1, alpha = 1, beta = 1,
    param = c(mu, sigma, alpha, beta))
pnl(q, mu = 0, sigma = 1, alpha = 1, beta = 1,
    param = c(mu, sigma, alpha, beta))
qnl(p, mu = 0, sigma = 1, alpha = 1, beta = 1,
    param = c(mu, sigma, alpha, beta),
    tol = 10^-5, nInterpol = 100, subdivisions = 100, ...)
rnl(n, mu = 0, sigma = 1, alpha = 1, beta = 1,
    param = c(mu, sigma, alpha, beta))
```

Arguments

- **x, q** Vector of quantiles.
- **p** Vector of probabilities.
- **n** Number of random variates to be generated.
- **mu** Location parameter $\mu$, default is 0.
- **sigma** Scale parameter $\sigma$, default is 1.
alpha  Skewness parameter $\alpha$, default is 1.
beta   Shape parameter $\beta$, default is 1.
param Specifying the parameters as a vector of the form $c(\mu, \sigma, \alpha, \beta)$.
tol    Specified level of tolerance when checking if parameter beta is equal to 0.
subdivisions The maximum number of subdivisions used to integrate the density and determine the accuracy of the distribution function calculation.
nInterpol Number of points used in qnl for cubic spline interpolation of the distribution function.
...   Passes arguments to uniroot.

Details

Users may either specify the values of the parameters individually or as a vector. If both forms are specified, then the values specified by the vector param will overwrite the other ones.

The density function is

$$f(y) = \frac{\alpha \beta}{\alpha + \beta} \phi \left( \frac{y - \mu}{\sigma} \right) \left[ R \left( \alpha \sigma - \frac{(y - \mu)}{\sigma} \right) + R \left( \beta \sigma + \frac{(y - \mu)}{\sigma} \right) \right]$$

The distribution function is

$$F(y) = \Phi \left( \frac{y - \mu}{\sigma} \right) - \phi \left( \frac{y - \mu}{\sigma} \right) \left[ \beta R(\alpha\sigma - \frac{y - \mu}{\sigma}) - \alpha R(\beta\sigma + \frac{y - \mu}{\sigma}) \right] / (\alpha + \beta)$$

The function $R(z)$ is the Mills’ Ratio, see millsR.

Generation of random observations from the normal Laplace distribution using rnl is based on the representation

$$Y \sim Z + W$$

where $Z$ and $W$ are independent random variables with

$$Z \sim N(\mu, \sigma^2)$$

and $W$ following an asymmetric Laplace distribution with pdf

$$f_W(w) = \begin{cases} 
(\alpha \beta)/(\alpha + \beta)e^{\beta w} & \text{for } w \leq 0 \\
(\alpha \beta)/(\alpha + \beta)e^{-\beta w} & \text{for } w > 0
\end{cases}$$

Value

dnl gives the density function, pnl gives the distribution function, qnl gives the quantile function and rnl generates random variates.

Author(s)

David Scott <d.scott@auckland.ac.nz>, Jason Shicong Fu
References


Examples

```r
param <- c(0,1,3,2)
par(mfrow = c(1,2))

## Curves of density and distribution
curve(dnl(x, param = param), -5, 5, n = 1000)
title("Density of the Normal Laplace Distribution")
curve(pnl(x, param = param), -5, 5, n = 1000)
title("Distribution Function of the Normal Laplace Distribution")

## Example of density and random numbers
par(mfrow = c(1,1))
param1 <- c(0,1,1,1)
data1 <- rnl(1000, param = param1)
curve(dnl(x, param = param1),
from = -5, to = 5, n = 1000, col = 2)
hist(data1, freq = FALSE, add = TRUE)
title("Density and Histogram")
```

Description

Functions to calculate the mean, variance, skewness and kurtosis of a specified normal Laplace distribution.

Usage

```r
nlMean(mu = 0, sigma = 1, alpha = 1, beta = 1, param = c(mu, sigma, alpha, beta))
nlVar(mu = 0, sigma = 1, alpha = 1, beta = 1, param = c(mu, sigma, alpha, beta))
nlSkew(mu = 0, sigma = 1, alpha = 1, beta = 1, param = c(mu, sigma, alpha, beta))
nlKurt(mu = 0, sigma = 1, alpha = 1, beta = 1, param = c(mu, sigma, alpha, beta))
```
Arguments

- **mu**: Location parameter $\mu$, default is 0.
- **sigma**: Scale parameter $\sigma$, default is 1.
- **alpha**: Skewness parameter $\alpha$, default is 1.
- **beta**: Shape parameter $\beta$, default is 1.
- **param**: Specifying the parameters as a vector of the form $c(mu, sigma, alpha, beta)$.

Details

Users may either specify the values of the parameters individually or as a vector. If both forms are specified, then the values specified by the vector `param` will overwrite the other ones.

The mean function is

$$E(Y) = \mu + \frac{1}{\alpha} - \frac{1}{\beta}.$$ 

The variance function is

$$V(Y) = \sigma^2 + \frac{1}{\alpha^2} + \frac{1}{\beta^2}.$$ 

The skewness function is

$$\gamma = \frac{[2/\alpha^3 - 2/\beta^3] / [\sigma^2 + 1/\alpha^2 + 1/\beta^2]^{3/2}}{}.$$ 

The kurtosis function is

$$\Gamma = \frac{[6/\alpha^4 + 6/\beta^4] / [\sigma^2 + 1/\alpha^2 + 1/\beta^2]^2}{}.$$ 

Value

`nlMean` gives the mean of the skew hyperbolic `nlVar` the variance, `nlskew` the skewness, and `nlkurt` the kurtosis.

Author(s)

David Scott <d.scott@auckland.ac.nz>, Jason Shicong Fu

References


Examples

```r
param <- c(10, 1, 5, 9)
nlMean(param = param)
nlVar(param = param)
nlskew(param = param)
nlkurt(param = param)
```

curve(dnl(x, param = param), -10, 10)
summary.nlFit  

Summarizing Normal Laplace Distribution Fit

Description

summary Method for class "nlFit".

Usage

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

Arguments

object      An object of class "nlFit", resulting from a call to nlFit.
x          An object of class "summary.nlFit" resulting from a call to summary.nlFit.
digits      The number of significant digits to use when printing.
...       Further arguments passed to or from other methods.

Details

summary.nlFit calculates standard errors for the estimates of \( \mu \), \( \sigma \), \( \alpha \), and \( \beta \) of the normal laplace distribution parameter vector \( \text{param} \) if the Hessian from the call to nlFit is available.

Value

If the Hessian is available, summary.nlFit computes standard errors for the estimates of \( \mu \), \( \sigma \), \( \alpha \), and \( \beta \), and adds them to object as object$sds. Otherwise, no calculations are performed and the composition of object is unaltered.

summary.nlFit invisibly returns object with class changed to summary.nlFit.

See also

nlFit, summary.nlFit, print.summary.nlFit

See Also

nlFit, summary.
Examples

```r
# Continuing the nlFit() example:
param <- c(2, 2, 1, 1)
dataVector <- rnl(500, param = param)
fit <- nlFit(dataVector, hessian = TRUE)
print(fit)
summary(fit)
```
Index

*Topic distribution
  nlCheckPars, 3
  nlFit, 4
  nlFitStart, 7
  nlPlots, 9
  NormalLaplaceDistribution, 10
  NormalLaplaceMeanVar, 12
  summary.nlFit, 14
*Topic math
  MillsRatio, 2
*Topic nlplot
  nlPlots, 9
*Topic package
  NormalLaplace-package, 2
coeff.nlFit(nlFit), 4
dnl, 2, 7, 8, 10
dnl(NormalLaplaceDistribution), 10
hist, 6–8
logHist, 7
millsR, 2, 11
millsR(MillsRatio), 2
MillsRatio, 2

nlCheckPars, 3
nlFit, 4, 8, 10, 14
nlFitStart, 6, 7, 7
nlFitStartMoM(nlFitStart), 7
nlKurt(NormalLaplaceMeanVar), 12
nlm, 6, 7
nlMean(NormalLaplaceMeanVar), 12
nlminb, 6
nlPlots, 9
nlSkew(NormalLaplaceMeanVar), 12
nlVar(NormalLaplaceMeanVar), 12
NormalLaplace(NormalLaplace-package), 2
NormalLaplace-package, 2

NormalLaplaceDistribution, 10
NormalLaplaceMeanVar, 2, 12

optim, 5–8
par, 5, 7
plot.nlFit(nlFit), 4
pn1(NormalLaplaceDistribution), 10
ppn1, 7
ppn1(nlPlots), 9
ppoints, 10
print.nlFit, 14
print.nlFit(nlFit), 4
print.summary.nlFit(summary.nlFit), 14
qqn1(NormalLaplaceDistribution), 10
qqn1(nlPlots), 9
rn1(NormalLaplaceDistribution), 10
summary, 14
summary.nlFit, 14
vcov.nlFit(nlFit), 4