Package ‘logitnorm’

January 7, 2021

Title Functions for the Logitnormal Distribution
Version 0.8.38
Author Thomas Wutzler
Maintainer Thomas Wutzler <twutz@bgc-jena.mpg.de>
Description Density, distribution, quantile and random generation function for the logitnormal distribution. Estimation of the mode and the first two moments. Estimation of distribution parameters.
Depends
Suggests RUnit, knitr, markdown, ggplot2, reshape2
VignetteBuilder knitr
License GPL-2
LazyData true
RoxygenNote 7.1.1
NeedsCompilation no
Repository CRAN
Date/Publication 2021-01-07 07:00:20 UTC

R topics documented:

logitnorm-package ........................................... 2
dlogitnorm ............................................. 3
invlogit ................................................... 4
logit ...................................................... 4
modeLogitnorm ........................................... 5
momentsLogitnorm ....................................... 6
plogitnorm ................................................. 7
qlogitnorm ................................................ 7
rlogitnorm ................................................. 8
twCoefLogitnorm ....................................... 9
twCoefLogitnormCI ...................................... 10
twCoefLogitnormE ...................................... 11
twCoefLogitnormMLE ................................... 12
logitnorm-package

Utilities for the logitnormal distribution in R

Description

Utilities for the logitnormal distribution in R

- Density, distribution, quantile and random generation function.
- Estimation of the mode and the first two moments.
- Estimation of distribution parameters from observations.

Details

The package provides the main distribution functions:

- density \texttt{dlogitnorm},
- distribution \texttt{plogitnorm},
- quantile \texttt{qlogitnorm}, and
- random generation function \texttt{rlogitnorm}.

Transformation functions

- (0,1) \rightarrow (-\infty,\infty): \texttt{logit}
- (-\infty,\infty) \rightarrow (0,1): \texttt{invlogit}

Moments and mode

- Expected value and variance: \texttt{momentsLogitnorm}
- Mode: \texttt{modeLogitnorm}

Estimating parameters

- from mode and upper quantile: \texttt{twCoefLogitnormMLE}
- from mode and constraint to be unimodal and maximally flat: \texttt{twCoefLogitnormMLEFlat}
- from median and upper quantile: \texttt{twCoefLogitnorm}
- from expected value, i.e. mean and upper quantile: \texttt{twCoefLogitnormE}
- from a confidence interval which is symmetric at normal scale: \texttt{twCoefLogitnormCi}
- from prescribed quantiles: \texttt{twCoefLogitnormN}

Have a look at the package vignettes.
Author(s)
Thomas Wutzler

References

Description
Density function of logitnormal distribution

Usage
dlogitnorm(x, mu = 0, sigma = 1, log = FALSE, ...)

Arguments
x vector of quantiles
mu distribution parameters
sigma
log if TRUE, the log-density is returned
... further arguments passed to dnorm: mean, and sd for mu and sigma respectively.

Details
Logitnorm distribution
• density function: dlogitnorm
• distribution function: plogitnorm
• quantile function: qlogitnorm
• random generation function: rlogitnorm

The function is only defined in interval (0,1), but the density returns 0 outside the support region.

Author(s)
Thomas Wutzler

See Also
logitnorm
Description
Transforming (-Inf,Inf) to original scale (0,1)

Usage
invlogit(q, ...)

Arguments
q
...

Details
function \( f(z) = \frac{e^z}{e^z+1} = \frac{1}{1+e^{-z}} \)

Author(s)
Thomas Wutzler

See Also
logit
logitnorm

Description
Transforming (0,1) to normal scale (-Inf Inf)

Usage
logit(p, ...)

Arguments
p
...

Details

function $\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = \log(p) - \log(1-p)$

Author(s)

Thomas Wutzler

See Also

invlogit
logitnorm

Description

Mode of the logitnormal distribution by numerical optimization

Usage

modeLogitnorm(mu, sigma, tol = invlogit(mu)/1000)

Arguments

mu parameter mu
sigma parameter sigma
tol precisions of the estimate

Author(s)

Thomas Wutzler

See Also

logitnorm
momentsLogitnorm

---

momentsLogitnorm  momentsLogitnorm

---

Description

First two moments of the logitnormal distribution by numerical integration

Usage

momentsLogitnorm(mu, sigma, abs.tol = 0, ...

Arguments

mu  parameter mu
sigma  parameter sigma
abs.tol  changing default to integrate
...  further parameters to the integrate function

Value

named numeric vector with components

- mean: expected value, i.e. first moment
- var: variance, i.e. second moment

Author(s)

Thomas Wutzler

Examples

(res <- momentsLogitnorm(4,1))
(res <- momentsLogitnorm(5,0.1))
plogitnorm

Description
Distribution function for logitnormal distribution

Usage
plogitnorm(q, mu = 0, sigma = 1, ...)

Arguments
- q: vector of quantiles
- mu: distribution parameters
- sigma
- ...

Author(s)
Thomas Wutzler

See Also
- logitnorm

qlogitnorm

Description
Quantiles of logitnormal distribution.

Usage
qlogitnorm(p, mu = 0, sigma = 1, ...)

Arguments
- p: vector of probabilities
- mu: distribution parameters
- sigma
- ...
Author(s)

Thomas Wutzler

See Also

logitnorm

Description

Random number generation for logitnormal distribution

Usage

rlogitnorm(n, mu = 0, sigma = 1, ...)

Arguments

n number of observations
mu distribution parameter
sigma distribution parameter
... arguments to rnorm

Author(s)

Thomas Wutzler

See Also

logitnorm
Description
Estimating coefficients of logitnormal distribution from median and upper quantile

Usage
_twCoefLogitnorm(median, quant, perc = 0.975, ...)

Arguments
- median: numeric vector: the median of the density function
- quant: numeric vector: the upper quantile value
- perc: numeric vector: the probability for which the quantile was specified
...

Value
numeric matrix with columns c("mu","sigma") rows correspond to rows in median, quant, and perc

Author(s)
Thomas Wutzler

See Also
logitnorm

Examples
# estimate the parameters, with median at 0.7 and upper quantile at 0.9
med = 0.7; upper = 0.9
med = 0.2; upper = 0.4
(theta <- _twCoefLogitnorm(med,upper))

x <- seq(0,1,length.out = 41)[-c(1,41)] # plotting grid
px <- plogitnorm(x,mu = theta[1],sigma = theta[2]) # percentiles function
plot(px~x); abline(v = c(med,upper),col = "gray"); abline(h = c(0.5,0.975),col = "gray")

dx <- dlogitnorm(x,mu = theta[1],sigma = theta[2]) # density function
plot(dx~x); abline(v = c(med,upper),col = "gray")

# vectorized
(theta <- _twCoefLogitnorm(seq(0.4,0.8,by = 0.1),0.9))
\texttt{twCoefLogitnormCi} <- \texttt{function()}{
  \# xr = rlogitnorm(1e5, mu = theta["mu"], sigma = theta["sigma"])
  \# median(xr)
  invlogit(theta["mu"])
  qlogitnorm(0.975, mu = theta["mu"], sigma = theta["sigma"])
}\n
\textbf{twCoefLogitnormCi} \hspace{1cm} \textit{twCoefLogitnormCi}

\textbf{Description}

Calculates mu and sigma of the logitnormal distribution from lower and upper quantile, i.e. confidence interval.

\textbf{Usage}

\texttt{twCoefLogitnormCi(lower, upper, perc = 0.975, sigmaFac = qnorm(perc), isTransScale = FALSE)}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{lower} \hspace{1cm} value at the lower quantile, i.e. practical minimum
  \item \texttt{upper} \hspace{1cm} value at the upper quantile, i.e. practical maximum
  \item \texttt{perc} \hspace{1cm} numeric vector: the probability for which the quantile was specified
  \item \texttt{sigmaFac} \hspace{1cm} sigmaFac = 2 is 95\% sigmaFac = 2.6 is 99\% interval
  \item \texttt{isTransScale} \hspace{1cm} if true lower and upper are already on logit scale
\end{itemize}

\textbf{Value}

named numeric vector: mu and sigma parameter of the logitnormal distribution.

\textbf{Author(s)}

Thomas Wutzler

\textbf{See Also}

\texttt{logitnorm}
Examples

```
mu = 2
sd = c(1,0.8)
p = 0.99
lower <- l <- qlogitnorm(1 - p, mu, sd)  # p-confidence interval
upper <- u <- qlogitnorm(p, mu, sd)    # p-confidence interval
cf <- twCoefLogitnormCi(lower,upper, perc = p)
all.equal(cf[,"mu"], c(mu,mu) )
all.equal(cf[,"sigma"], sd)
```

Description

Estimating coefficients of logitnormal distribution from expected value, i.e. mean, and upper quantile.

Usage

```
twCoefLogitnormE(mean, quant, perc = c(0.975),
    method = "BFGS", theta0 = c(mu = 0, sigma = 1),
    returnDetails = FALSE, ...)
```

Arguments

- `mean` the expected value of the density function
- `quant` the quantile values
- `perc` the probabilities for which the quantiles were specified
- `method` method of optimization (see `optim`)
- `theta0` starting parameters
- `returnDetails` if TRUE, the full output of optim is returned with attribute resOptim
  ...

Value

named numeric matrix with estimated parameters of the logitnormal distribution. colnames: c("mu","sigma")

Author(s)

Thomas Wutzler

See Also

`logitnorm`
Examples

# estimate the parameters
(thetaE <- twCoefLogitnormE(0.7, 0.9))

x <- seq(0, 1, length.out = 41)[-c(1, 41)]  # plotting grid
px <- plogitnorm(x, mu = thetaE[1], sigma = thetaE[2])  # percentiles function
plot(px ~ x); abline(v = c(0.7, 0.9), col = "gray"); abline(h = c(0.5, 0.975), col = "gray")
dx <- dlogitnorm(x, mu = thetaE[1], sigma = thetaE[2])  # density function
plot(dx ~ x); abline(v = c(0.7, 0.9), col = "gray")

z <- rlogitnorm(1e5, mu = thetaE[1], sigma = thetaE[2])
mean(z)  # about 0.7

# vectorized
(theta <- twCoefLogitnormE(mean = seq(0.4, 0.8, by = 0.1), quant = 0.9))

Description

Estimating coefficients of logitnormal distribution from mode and upper quantile

Usage

twCoefLogitnormMLE(mle, quant, perc = 0.999)

Arguments

mle    numeric vector: the mode of the density function
quant  numeric vector: the upper quantile value
perc   numeric vector: the probability for which the quantile was specified

Value

numeric matrix with columns c("mu", "sigma") rows correspond to rows in mle, quant, and perc

Author(s)

Thomas Wutzler

See Also

logitnorm
Examples

# estimate the parameters, with mode 0.7 and upper quantile 0.9
mode = 0.7; upper = 0.9
mode = 0.2; upper = 0.7
#mode = 0.5; upper = 0.9
(theta <- twCoefLogitnormMLE(mode, upper))
x <- seq(0,1,length.out = 41)[-c(1,41)] # plotting grid
px <- plogitnorm(x,mu = theta[1],sigma = theta[2]) # percentiles function
plot(px~x); abline(v = c(mode,upper),col = "gray"); abline(h = c(0.999),col = "gray")
dx <- dlogitnorm(x,mu = theta[1],sigma = theta[2]) # density function
plot(dx~x); abline(v = c(mode,upper),col = "gray")
# vectorized
(theta <- twCoefLogitnormMLE(mle = seq(0.4,0.8,by = 0.1), quant = upper))
# flat
(theta <- twCoefLogitnormMLEFlat(mode))

description

Estimating coefficients of a maximally flat unimodal logitnormal distribution from mode

Usage

twCoefLogitnormMLEFlat(mle)

Arguments

mle numeric vector: the mode of the density function

Details

When increasing the sigma parameter, the distribution becomes eventually becomes bi-model, i.e. has two maxima. This function estimates parameters for given mode, so that the distribution assigns high density to a maximum range, i.e. is maximally flat, but still is unimodal.

Author(s)

Thomas Wutzler
twCoefLogitnormN

Description

Estimating coefficients from a vector of quantiles and percentiles (non-vectorized).

Usage

twCoefLogitnormN(quant, perc = c(0.5, 0.975),
method = "BFGS", theta0 = c(mu = 0, sigma = 1),
returnDetails = FALSE, ...)

Arguments

- `quant`: the quantile values
- `perc`: the probabilities for which the quantiles were specified
- `method`: method of optimization (see `optim`)
- `theta0`: starting parameters
- `returnDetails`: if TRUE, the full output of optim is returned instead of only entry par
- `...`: further parameters passed to optim, e.g. `control = list(maxit = 1000)`

Value

named numeric vector with estimated parameters of the logitnormal distribution. names: c("mu","sigma")

Author(s)

Thomas Wutzler

See Also

- `logitnorm`

Examples

# experiment of re-estimation the parameters from generated observations
thetaTrue <- c(mu = 0.8, sigma = 0.7)
obsTrue <- rlogitnorm(thetaTrue["mu"],thetaTrue["sigma"], n = 500)
obs <- obsTrue + rnorm(100, sd = 0.05)  # some observation uncertainty
plot(density(obsTrue), col = "blue"); lines(density(obs))

# re-estimate parameters based on the quantiles of the observations
(theta <- twCoefLogitnorm( median(obs), quantile(obs,probs = 0.9), perc = 0.9))

# add line of estimated distribution
x <- seq(0,1,length.out = 41)[-c(1,41)]  # plotting grid
dx <- dlogitnorm(x,mu = theta[1],sigma = theta[2])
lines( dx ~ x, col = "orange")
Description

Estimating coefficients of logitnormal distribution from mode and given mu

Usage

twSigmaLogitnorm(mle, mu = 0)

Arguments

mle numeric vector: the mode of the density function
mu for mu = 0 the distribution will be the flattest case (maybe bimodal)

Details

For a mostly flat unimodal distribution use twCoefLogitnormMLE(mle,0)

Value

numeric matrix with columns c("mu","sigma") rows correspond to rows in mle and mu

Author(s)

Thomas Wutzler

See Also

logitnorm

Examples

mle <- 0.8
(theta <- twSigmaLogitnorm(mle))
# x <- seq(0,1,length.out = 41)[-c(1,41)] # plotting grid
px <- plogitnorm(x,mu = theta[1],sigma = theta[2]) #percentiles function
plot(px-x); abline(v = c(mle),col = "gray")
dx <- dlogitnorm(x,mu = theta[1],sigma = theta[2]) #density function
plot(dx-x); abline(v = c(mle),col = "gray")
# vectorized
(theta <- twSigmaLogitnorm(mle = seq(0.401,0.8,by = 0.1)))
Index

* package
  logitnorm-package, 2

  dlogitnorm, 2, 3
  dnorm, 3

  integrate, 6
  invlogit, 2, 4, 5

  logit, 2, 4, 4
  logitnorm, 3–5, 7–12, 14, 15
  logitnorm (logitnorm-package), 2
  logitnorm-package, 2

  modelLogitnorm, 2, 5
  momentsLogitnorm, 2, 6

  optim, 11, 14

  plogitnorm, 2, 3, 7

  qlogitnorm, 2, 3, 7

  rlogitnorm, 2, 3, 8
  rnorm, 8

  twCoefLogitnorm, 2, 9
  twCoefLogitnormCi, 2, 10
  twCoefLogitnormE, 2, 11
  twCoefLogitnormMLE, 2, 12, 15
  twCoefLogitnormMLEFlat, 2, 13
  twCoefLogitnormN, 2, 14
  twSigmaLogitnorm, 15