Package 'normalp'

July 22, 2025

Version 0.7.2.1

Date 2020-02-14

Title Routines for Exponential Power Distribution

Author Angelo M. Mineo <angelo.mineo@unipa.it>

Maintainer Angelo M. Mineo <angelo.mineo@unipa.it>

Depends R (>= 1.5.0)

Description
A collection of utilities referred to Exponential Power distribution, also known as General Error Distribution (see Mineo, A.M. and Ruggieri, M. (2005), A software Tool for the Exponential Power Distribution: The normalp package. In Journal of Statistical Software, Vol. 12, Issue 4).

License GPL

URL https://www.r-project.org,
https://www.unipa.it/persone/docenti/m/angelo.mineo

NeedsCompilation no

Repository CRAN

Date/Publication 2023-12-11 08:22:42 UTC

Contents

`normalp-package` *Package for exponential power distributions (EPD)*

Description

This package implements a collection of utilities referred to exponential power distributions, also known as General Error Distribution. These utilities have been developed by some researcher of the University of Palermo, Italy.

Details

Package:	normalp
Type:	Package
Version:	0.7.2
Date:	2020-02-14
License:	GPL

Author(s)

Angelo M. Mineo
Maintainer: Angelo M. Mineo <angelo.mineo@unipa.it>

References

Chiodi, M. (1986) *Procedures for generating pseudo-random numbers from a normal distribution of order p*, Rivista di Statistica Applicata, 19, pp. 7-26.

Mineo, A. (1989) *The norm-p estimation of location, scale and simple linear regressione*, Lectures Notes in Statistics: Statistical Modelling, 57, pp. 222-233.

Mineo, A.M. (1994) *Un nuovo metodo di stima di p per una corretta valutazione dei parametri di intensita e di scala di una curva normale di ordine p*, Atti della XXXVII Riunione Scientifica della Societa Italiana di Statistica, San Remo, Vol. 2, pp. 147-154.

Mineo, A.M. (2003) *On the Estimation of the Structure Parameter of a Normal Distribution of Order p*, *Statistica*, anno LXIII, n. 1, pp. 109-122.

dnormp *Density function of an exponential power distribution*

Description

Density function for the exponential power distribution with location parameter mu, scale parameter sigmap and shape parameter p.

Usage

```
dnormp(x, mu=0, sigmap=1, p=2, log=FALSE)
```

Arguments

x	Vector of quantiles.
mu	Vector of location parameters.
sigmap	Vector of scale parameters.
p	Shape parameter.
log	Logical; if TRUE, the density is given as log(density).

Details

If mu, sigmap or p are not specified they assume the default values 0, 1 and 2, respectively. The exponential power distribution has density function

$$f(x) = \frac{1}{2p^{(1/p)}\Gamma(1+1/p)\sigma_p} e^{-\frac{|x-\mu|^p}{p\sigma_p^p}}$$

where μ is the location parameter, σ_p the scale parameter and p the shape parameter. When $p = 2$ the exponential power distribution becomes the Normal Distribution, when $p = 1$ the exponential power distribution becomes the Laplace Distribution, when $p \to \infty$ the exponential power distribution becomes the Uniform Distribution.

Value

dnormp gives the density function of an exponential power distribution.

Author(s)

Angelo M. Mineo

See Also

Normal for the Normal distribution, Uniform for the Uniform distribution, and Special for the Gamma function.

Examples

```
## Compute the density for a vector x with mu=0, sigmap=1 and p=1.5
## At the end we have the graph of the exponential power distribution
## density function with p=1.5
x <- c(-1, 1)
f <- dnormp(x, p=1.5)
print(f)
plot(function(x) dnormp(x, p=1.5) , -4, 4,
        main = "Exponential power distribution density function (p=1.5)", ylab="f(x)")
```

`estimatep` *Estimation of p*

Description

The `estimatep` function estimates the shape parameter p from a vector of observations.

Usage

```
estimatep(x, mu, p=2, method=c("inverse","direct"))
```

Arguments

`x`	Vector of observations.
`mu`	An estimate of the location parameter.
`p`	Starting value of the shape parameter.
`method`	Method used to estimate p from a sample.

Details

The used algorithm is based on a method proposed by A.M. Mineo (1994), which uses a particular index of kurtosis, called VI

$$VI = \frac{\sqrt{\Gamma(1/p)\Gamma(3/p)}}{\Gamma(2/p)}.$$

With `method` the user can choice between an inverse interpolation (faster) or a direct solution of the equation

$$\hat{VI} = \frac{\sqrt{\Gamma(1/p)\Gamma(3/p)}}{\Gamma(2/p)}.$$

Value

An estimate of p from a sample of observations.

Author(s)

Angelo M. Mineo

References

Mineo, A.M. (1994) *Un nuovo metodo di stima di p per una corretta valutazione dei parametri di intensita e di scala di una curva normale di ordine p*, Atti della XXXVII Riunione Scientifica della Societa Italiana di Statistica, San Remo, Vol. 2, pp. 147-154.

Examples

```
x<-rnormp(300,mu=1,sigmap=2,p=4)
p<-estimatep(x,mu=1,p=2)
p
```

---

graphnp — *Plot of exponential power distributions*

Description

The function `graphnp` returns on the same device, marked with different colours, from one to five exponential power distributions.

Usage

```
graphnp(p=c(1,2,3,4,5), mu=0, sigmap=1, title="Exponential Power Distributions")
```

Arguments

`p`	A vector of p values. His length must be maximum five.
`mu`	Value of the location parameter.
`sigmap`	Value of the scale parameter.
`title`	The title of the plot.

Details

If one or more values of `p` are greater than or equal to 50, `graphnp` will plot the density function of an uniform distribution.

Value

A graphic device with till five different curves. The curves have different colours and the device is completed by a legend.

Author(s)

Angelo M. Mineo

Examples

```
## Plot four different curves with p=1,2,3,4
## and 50 (it will plot an uniform distribution)
graphnp(c(1:4,50))
```

`kurtosis` *Indices of kurtosis*

Description

This function computes the theoretical and empirical values of three indices of kurtosis.

Usage

```
kurtosis(x = NULL, p, value = c("estimate", "parameter"))
```

Arguments

`x`	A sample of observations.
`p`	the shape parameter.
`value`	If is set to `estimate`, evaluate the indices using an estimate of p. Otherwise, if is set to `parameter` it uses the value specified in `p`.

Value

It returns the vector of the three indices of kurtosis VI, β_2 and β_p. Giving a vector as argument, it returns the estimates of the three indices, computed on the sample. On the other hand, giving the value of the shape parameter p, it returns the theoretical indices.

Author(s)

Angelo M. Mineo

References

Mineo, A.M. (1996) *La migliore combinazione delle osservazioni: curve normali di ordine p e stimatori di norma Lp*. PhD thesis.

Examples

```
kurtosis(p=2)
x<-rnormp(50,mu=0,sigmap=2,p=1.5)
kurtosis(x,p=2)
```

lmp — *Fitted linear model with exponential power distribution errors*

Description

The function `lmp` is used to fit linear model. It can be used when the errors are distributed as an exponential power distribution.

Usage

```
lmp(formula, data, p)
```

Arguments

`formula`	A symbolic description of the model to be fitted.
`data`	An optional data frame containing the variables in the model. By default the variables are taken from the environment.
`p`	The shape parameter. If specified, this function estimates the parameter by using the $L_p - norm$ method.

Details

To evaluate the coefficients of the linear model, `lmp` uses the maximum likelihood estimators. This function can give some problems if the number of regressors is too high.

Value

The function `lmp` returns an object of `class` `"lmp"` and `"lm"`. The function `summary` print a summary of the results. The generic accessor functions `coefficients`, `effects`, `fitted.values` and `residuals` extract various useful features of the value returned by `lmp`. An object of `class` `"lmp"` is a list containing at least the following components:

`coefficients`	A named vector of coefficients.
`residuals`	The residuals, that is responses minus fitted values.
`fitted.values`	The fitted values.
`rank`	The numeric rank of the fitted linear model.
`df.residual`	The residual degrees of freedom computed as in `lm`.
`call`	The matched call.
`terms`	The `terms` object used.
`p`	Estimate of the shape parameter computed on residuals.
`knp`	A logical parameter used by `summary`.
`model`	The model frame used.
`iter`	If its value is 1 we have had a difficult convergence.

Author(s)

Angelo M. Mineo

References

Mineo, A.M. (1995) *Stima dei parametri di regressione lineare semplice quando gli errori seguono una distribuzione normale di ordine p (p incognito)*. Annali della Facolt\'a di Economia dell'Universit\'a di Palermo (Area Statistico-Matematica), pp. 161-186.

Examples

```
e<-rnormp(n=100,mu=0,sigmap=4,p=3,method="d")
x<-runif(100)
y<-0.5+2*x+e
lmp(y~x)
```

`paramp` *Estimation of location and scale parameters*

Description

The function `paramp` returns a list with five elements: arithmetic mean, M_p, standard deviation, S_p, and shape parameter p, estimated on a sample.

Usage

```
paramp(x, p)
```

Arguments

`x`	A vector of observations.
`p`	If specified, the algorithm uses this value for `p`, i.e. the algorithm does not use an estimate of `p`.

Value

The estimation of μ and p is based on an iterative method. To show differences between the least squares method and the maximum likelihood method, it prints out also the mean and the standard deviation.

`Mean`	Arithmetic mean.
`Mp`	The estimated value of the location parameter.
`Sd`	Standard deviation.
`Sp`	The estimated value of the scale parameter.
`p`	The estimated value of the shape parameter.
`iter`	If its value is 1, we have had problems on convergence.

Author(s)

Angelo M. Mineo

References

Mineo, A.M. (1996) *La migliore combinazione delle osservazioni: curve normali di ordine p e stimatori di norma Lp*. PhD thesis.

Examples

```
x<-rnormp(1000,2,3,4.2)
paramp(x)
```

---

plot.lmp — *Diagnostic plots for a lmp object*

Description

This function produces four plots: a plot of residuals against fitted value, a Normal Q-Q plot, an Exponential Power Distribution Q-Q plot, a Scale-Location plot, with a p-root of the standardized residuals against the fitted values.

Usage

```
## S3 method for class 'lmp'
plot(x, ...)
```

Arguments

`x`	A `lmp` object, typically result of `lmp`.
`...`	Further arguments passed to or from other methods.

Details

The standardized residuals in the Normal Q-Q plot are those of an object `lm`; in the Exponential Power distribution Q-Q plot and in the scale location plot the standardized residuals are computed as $(e_i - m_p)/(s_p)$.

Author(s)

Angelo M. Mineo

Examples

```
x<-1:20
z<-runif(20)
e<-rnormp(20,mu=0,sigmap=1,p=3)
y<-0.5+x+z+e
lmp.res<-lmp(y~x+z)
plot(lmp.res)
```

---

plot.simul.lmp — *Plots of the results of a simulation plan on a linear regression model*

Description

It returns the histograms of the estimates of the regression coefficients, of the scale parameter σ_p and of the shape parameter p.

Usage

```
## S3 method for class 'simul.lmp'
plot(x,...)
```

Arguments

x	A simul.lmp object, typically result of simul.lmp
...	Further arguments passed to or from other methods

Value

The histograms of all the coefficients of the linear regression model and of the estimates of the scale parameter σ_p and of the structure parameter p.

Author(s)

Angelo M. Mineo

Examples

```
sim<-simul.lmp(n=10,m=50,q=1,data=1.5,int=0,sigmap=1,p=3.5)
plot(sim)
```

plot.simul.mp	*Plots of the results of a simulation plan on the parameters of an exponential power distribution*

Description

It returns the histograms of the vector of means, estimates of μ, standard deviations, estimates of σ_p and estimates of p.

Usage

```
## S3 method for class 'simul.mp'
plot(x, ...)
```

Arguments

x	A simul.mp object, typically result of simul.mp
...	Further arguments passed to or from other methods

Value

The histograms of the estimates of the parameters of an exponential power distribution.

Author(s)

Angelo M. Mineo

Examples

```
## The histograms of all the computed estimates
a<-simul.mp(100,50,mu=0,sigmap=1,p=3)
plot(a)
```

pnormp	*Probability function of an exponential power distribution*

Description

Probability function for the exponential power distribution with location parameter mu, scale parameter sigmap and shape parameter p.

Usage

```
pnormp(q, mu=0, sigmap=1, p=2, lower.tail=TRUE, log.pr=FALSE)
```

Arguments

q	Vector of quantiles.
mu	Vector of location parameters.
sigmap	Vector of scale parameters.
p	Shape parameter.
lower.tail	Logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
log.pr	Logical; if TRUE, probabilities pr are given as $log(pr)$.

Details

If mu, sigmap or p are not specified they assume the default values 0, 1 and 2, respectively. The exponential power distribution has density function

$$f(x) = \frac{1}{2p^{(1/p)}\Gamma(1+1/p)\sigma_p} e^{-\frac{|x-\mu|^p}{p\sigma_p^p}}$$

where μ is the location parameter, σ_p the scale parameter and p the shape parameter. When $p = 2$ the exponential power distribution becomes the Normal Distribution, when $p = 1$ the exponential power distribution becomes the Laplace Distribution, when $p \to \infty$ the exponential power distribution becomes the Uniform Distribution.

Value

pnormp gives the probability of an exponential power distribution.

Author(s)

Angelo M. Mineo

See Also

Normal for the Normal distribution, Uniform for the Uniform distribution, and Special for the Gamma function.

Examples

```
## Compute the distribution function for a vector x with mu=0, sigmap=1 and p=1.5
## At the end we have the graph of the exponential power distribution function with p=1.5.
x <- c(-1, 1)
pr <- pnormp(x, p=1.5)
print(pr)
plot(function(x) pnormp(x, p=1.5), -4, 4,
          main = "Exponential Power Distribution Function (p=1.5)", ylab="F(x)")
```

qnormp — *Quantiles of an exponential power distribution*

Description

Quantiles for the exponential power distribution with location parameter `mu`, scale parameter `sigmap` and shape parameter `p`.

Usage

```
qnormp(pr, mu=0, sigmap=1, p=2, lower.tail=TRUE, log.pr=FALSE)
```

Arguments

`pr`	Vector of probabilities.
`mu`	Vector of location parameters.
`sigmap`	Vector of scale parameters.
`p`	Shape parameter.
`lower.tail`	Logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
`log.pr`	Logical; if TRUE, probabilities pr are given as $log(pr)$.

Details

If `mu`, `sigmap` or `p` are not specified they assume the default values 0, 1 and 2, respectively. The exponential power distribution has density function

$$f(x) = \frac{1}{2p^{(1/p)}\Gamma(1+1/p)\sigma_p} e^{-\frac{|x-\mu|^p}{p\sigma_p^p}}$$

where μ is the location parameter, σ_p the scale parameter and p the shape parameter. When $p = 2$ the exponential power distribution becomes the Normal Distribution, when $p = 1$ the exponential power distribution becomes the Laplace Distribution, when $p \rightarrow \infty$ the exponential power distribution becomes the Uniform Distribution.

Value

`qnormp` gives the quantiles of an exponential power distribution.

Author(s)

Angelo M. Mineo

See Also

`Normal` for the Normal distribution, `Uniform` for the Uniform distribution, and `Special` for the Gamma function.

Examples

```
## Compute the quantiles for a vector of probabilities x
## with mu=1, sigmap=2 and p=1.5
x <- 0.3
q <- qnormp(x, 1, 2, 1.5)
q
```

---

qqnormp *Quantile-Quantile plot for an exponential power distribution*

Description

The function `qqnormp` produces an exponential power distribution Q-Q plot of the values in `y`. The function `qqlinep` adds a line to an exponential power distribution Q-Q plot going through the first and the third quartile.

Usage

```
qqnormp(y, ylim, p, main, xlab, ylab, ...)
qqlinep(y, p=2, ...)
```

Arguments

`y`	Vector of observations.
`p`	The shape parameter.
`main, xlab, ylab`	Plot labels.
`ylim, ...`	Graphical parameters

Author(s)

Angelo M. Mineo

Examples

```
## Exponential power distribution Q-Q plot for a sample of 100 observations.
e<-rnormp(100,mu=0,sigmap=1,p=3)
qqnormp(e,p=3)
qqlinep(e,p=3)
```

`rnormp` *Pseudo-random numbers from an exponential power distribution*

Description

Generation of pseudo-random numbers from an exponential power distribution with location parameter `mu`, scale parameter `sigmap` and shape parameter `p`.

Usage

```
rnormp(n, mu = 0, sigmap = 1, p = 2, method = c("def", "chiodi"))
```

Arguments

`n`	Number of observations.
`mu`	Vector of location parameters.
`sigmap`	Vector of scale parameters.
`p`	Shape parameter.
`method`	If is set to the default method "`def`", it uses the method based on the transformation of a Gamma random variable. If set to "`chiodi`", it uses an algorithm based on a generalization of the Marsaglia formula to generate pseudo-random numbers from a normal distribution. The default method "`def`" is faster than the "`chiodi`" one (this one is introduced only for "historical" purposes).

Details

If `mu`, `sigmap` or `p` are not specified they assume the default values 0, 1 and 2, respectively. The exponential power distribution has density function

$$f(x) = \frac{1}{2p^{(1/p)}\Gamma(1+1/p)\sigma_p} e^{-\frac{|x-\mu|^p}{p\sigma_p^p}}$$

where μ is the location parameter, σ_p the scale parameter and p the shape parameter. When $p = 2$ the exponential power distribution becomes the Normal Distribution, when $p = 1$ the exponential power distribution becomes the Laplace Distribution, when $p \to \infty$ the exponential power distribution becomes the Uniform Distribution.

Value

`rnormp` gives a vector of n pseudo-random numbers from an exponential power distribution.

Author(s)

Angelo M. Mineo

References

Chiodi, M. (1986) *Procedures for generating pseudo-random numbers from a normal distribution of order p (p>1)*, Statistica Applicata, 1, pp. 7-26.

Marsaglia, G. and Bray, T.A. (1964) *A convenient method for generating normal variables*, SIAM rev., 6, pp. 260-264.

See Also

`Normal` for the Normal distribution, `Uniform` for the Uniform distribution, `Special` for the Gamma function and `.Random.seed` for the random number generation.

Examples

```
## Generate a random sample x from an exponential power distribution
## At the end we have the histogram of x
x <- rnormp(1000, 1, 2, 1.5)
hist(x, main="Histogram of the random sample")
```

---

simul.lmp — *Simulation planning for a linear regression model with errors distributed as an exponential power distribution*

Description

This function performs a Monte Carlo simulation to compare least squares estimators and Maximum Likelihood estimators for a linear regression model with errors distributed as an exponential power distribution. The regressors are drawn from an Uniform distribution.

Usage

```
simul.lmp(n, m, q, data, int=0, sigmap=1, p=2, lp=FALSE)
```

Arguments

`n`	Sample size.
`m`	Number of samples.
`q`	Number of regressors.
`data`	A vector of coefficients.
`int`	Value of the intercept.
`sigmap`	The scale parameter.
`p`	The shape parameter.
`lp`	Logical. If `TRUE`, it evaluates the coefficients with p known.

Value

The function `simul.lmp` returns an object of class `"simul.lmp"`. A component of this object is a table of means and variances of the m estimates of the regression coefficients and of the scale paramenter σ_p. The `summary` shows this table and the arguments of the simulation plan. The function `plot` returns the histograms of the computed estimates.

Author(s)

Angelo M. Mineo

References

Mineo, A.M. (1995) *Stima dei parametri di regressione lineare semplice quando gli errori seguono una distribuzione normale di ordine p (p incognito)*. Annali della Facolt\'a di Economia dell'Universit\'a di Palermo (Area Statistico-Matematica), pp. 161-186.

Examples

```
## Simulation of 50 samples of size 10 for a linear regression model with 1 regressor.
simul.lmp(10,50,1,data=1.5,int=1,sigmap=1,p=3,lp=FALSE)
```

`simul.mp` *Simulation planning for the parameters of an exponential power distribution*

Description

This function performs a Monte Carlo simulation to compare least square estimators and Maximum Likelihood estimators for the parameters of an exponential power distribution. For each sample, it calls the function `paramp`, returning the arithmetic means, the max-likelihood estimates of the location parameter, the standard deviations, the max-likelihood estimates of the scale parameter and the estimates of the shape parameter.

Usage

```
simul.mp(n, m, mu=0, sigmap=1, p=2)
```

Arguments

`n`	Sample size.
`m`	Number of samples.
`mu`	Value of the location parameter.
`sigmap`	Value of the scale parameter.
`p`	the shape parameter.

Value

This function is useful to compare several kinds of estimators. It returns an object of class "simul.mp", a list containing the following components:

dat	A matrix $m \times 5$ containing the results of paramp for each sample.
table	A matrix reporting the means and the variances of the values of the five estimators.

Author(s)

Angelo M. Mineo

References

Mineo, A.M. (1995) *Stima dei parametri di intensit\'a e di scala di una curva normale di ordine p (p incognito)*. Annali della Facolt\'a di Economia dell'Universit\'a di Palermo (Area Statistico-Matematica), pp. 125-159.

Examples

```
## Simulation plan for 100 samples of size 20, with mu=0, sigmap=1, p=3.
simul.mp(20,100,mu=0,sigmap=1,p=3)
```

---

summary.lmp *Summarize linear model fits with exponential power distribution errors*

Description

This function is the summary method for class "lmp". This function produces a set of results for a linear regression model. By assuming that in a linear regression model the errors are distributed as an exponential power distribution, we can use the function lmp.

Usage

```
## S3 method for class 'lmp'
summary(object, ...)
## S3 method for class 'summary.lmp'
print(x, ...)
```

Arguments

object	An object of class "lmp", a result of a call to lmp.
x	An object of class "summary.lmp".
...	Further arguments passed to or from other methods.

Value

The function `summary` returns a list of summary statistics of the fitted linear model given in `lmp`, using the components (list elements) `call` and `terms` from its argument, plus

`Call` The matched call.

`Residuals` A summary of the vector of residuals e_i.

`Coefficients` Vector of coefficients.

`Estimate of p` An estimate of the shape parameter p.

`Power deviation of order p`
The power deviation of order p given by

$$S_p = \left[\frac{\sum e_i^p}{n-q}\right]^{\frac{1}{p}}$$

where q is either the number of the estimated regression coefficients if p is known, either the number of the estimated regression coefficients plus 1 if p is estimated.

Author(s)

Angelo M. Mineo

Examples

```
x<-runif(30)
e<-rnormp(30,0,3,1.25)
y<-0.5+x+e
L<-lmp(y~x)
summary(L)
```

`summary.simul.lmp` *Summarize simulation results on linear regression model*

Description

This function is the `summary` method for class `"simul.lmp"`. This function produces a set of results for a simulation plan for a linear regression model with errors distributed as an exponential power distribution.

Usage

```
## S3 method for class 'simul.lmp'
summary(object, ...)
## S3 method for class 'summary.simul.lmp'
print(x, ...)
```

Arguments

`object`	An object of class "`simul.lmp`", a result of a call to `simul.lmp`.
`x`	An object of class "`summary.simul.lmp`", usually a result of a call to `summary.simul.lmp`.
`...`	Further arguments passed to or from other methods.

Value

This function returns this information:

`Results`	Table containing the simulation results.
`Coefficients`	The true values of coefficients used on the simulation model.
`Formula`	The used linear regression model.
`Number of samples`	Number of samples generated.
`Value of p`	Value of the shape parameter p used to draw the samples.
`Number of samples with problems on convergence`	If p is estimated, we have information on the number of samples with problems on convergence.

Author(s)

Angelo M. Mineo

Examples

```
ris<-simul.lmp(100,20,2,data=c(3,2),int=0,sigmap=1,p=3)
summary(ris)
```

Index