Package 'ecpdist'

September 5, 2024

Contents

decp   *Density function*

Description

Compute the density function of the extended Chen-Poisson (ecp) distribution.

Usage

```
decp(x, lambda, gamma, phi, log = FALSE)
```

Arguments

x	vector of quantiles.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.
log	logical value

Value

Numeric value of the density function.

If log = TRUE, numeric value of the logarithm of the function.

Examples

```
decp(2, 1, 1, 1, log = FALSE) # density function
```

---

ecp_kmoment *The k-th raw moment of extended Chen-Poisson distribution*

Description

Computes the k-th raw moment of the extended Chen-Poisson (ecp) distribution.

Usage

```
ecp_kmoment(k, lambda, gamma, phi)
```

Arguments

k	a positive integer.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.

Details

To obtain the value of k-th raw moment of the Extended Chen-Poisson distribution, it is necessary to use numerical integration. For that purpose, the R function 'integrate()' can be used, which returns the estimated value of the integral and also the integral absolute error (see details about function 'integrate()'). Therefore, to obtain the variance, the first component of each k-th raw moments must be selected.

Value

Estimated value of k-th raw moment, based on numerical integration, as well as integral absolute error obtained from the function integrate.

Examples

```
ecp_kmoment(k = 1, lambda = .1, gamma = .5, phi = - .2) # First raw moment.
ecp_kmoment(k = 2, lambda = .1, gamma = .5, phi = - .2) # Second raw moment.
ecp_kmoment(k = 2, lambda = .1, gamma = .5, phi = - .2)[1] -
ecp_kmoment(k = 1, lambda = .1, gamma = .5, phi = - .2)[1]^2 # Variance.
```

---

ecp_kmoment_cond   *The conditional k-th moment of extended Chen-Poisson distribution*

Description

Computes the conditional k-th moment of the extended Chen-Poisson (ecp) distribution.

Usage

```
ecp_kmoment_cond(x, k, lambda, gamma, phi)
```

Arguments

x	vector of quantiles.
k	a positive integer.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.

Details

The conditional k-th moment of a distribution is given by $E(X^k \mid X > x)$. To obtain this value for the Extended Chen-Poisson distribution, it is necessary to use numerical integration. For that purpose, the R function 'integrate()' can be used (see details about function 'integrate()'). Note that when $k = 1$ and $x = 0$, the conditional moment is equal to the expected value, i.e., $E(X \mid X > 0) = E(X)$.

Value

Estimated value of conditional k-th moment, based on numerical integration, using the function integrate.

Examples

```
ecp_kmoment_cond(x = 0, k = 1, lambda = .1, gamma = .5, phi = - .2)
```

---

ecp_mrl   *Mean residual life function*

Description

Computes the mean residual life function of the extended Chen-Poisson (ecp) distribution.

Usage

```
ecp_mrl(x, lambda, gamma, phi)
```

Arguments

x	vector of quantiles.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.

Value

Estimated value of mean residual life function, based on numerical integration.

Examples

```
ecp_mrl(x = 5, lambda = .1, gamma = .5, phi = - .2)
```

---

ecp_plot   *Plot function*

Description

Plots the density, cumulative distribution, hazard, cumulative hazard, survival and quantile functions of the extended Chen-Poisson (ecp) distribution.

Usage

```
ecp_plot(
  data_type,
  from = NULL,
  to = NULL,
  xlim = NULL,
  ylim = NULL,
  x = NULL,
  lambda,
  gamma,
```

```
  phi,
  log = FALSE,
  func_type,
  title,
  col = "black",
  lty = 1
)
```

Arguments

data_type	specifies whether the input is a x vector of data values or an expression. Possible types are
	- 'data' for data values,
	- 'expression' for expression.
from	lower x axis limit, by default from = 0.
to	upper x axis limit, by default to = 1.
xlim	x axis limits, by default xlim = c(from, to).
ylim	y axis limits.
x	vector of data values when data_type = "data".
lambda, gamma	parameter values > 0.
phi	parameter value != 0.
log	logical value.
func_type	specifies the type of function to be plotted. Possible types are
	- 'density' for density plot,
	- 'hazard' for hazard plot,
	- 'cumulative hazard' for cumulative hazard plot,
	- 'survival' for survival plot,
	- 'cumulative distribution' for cumulative distribution plot,
	- 'quantile' for quantile plot.
title	title of the graphic.
col	to set the color of the graphic.
lty	to set the line type.

Value

If log = TRUE, numeric value of the logarithm of the function.

If cum_haz = TRUE, numeric value of the cumulative hazard function.

graphic of the chosen ecp function.

Examples

```
#  Example of plotting cumulative distribution using an expression

ecp_plot(data_type = "expression", from = 0, to = 6, lambda = 2, gamma = 0.3,
phi = 30, func_type = "cumulative distribution",
title = "Cumulative Distribution Function (Expression)")

#  Example of plotting an unimodal hazard function using an expression

ecp_plot(data_type = "expression", lambda = 2, gamma = 0.3, phi = 30,
func_type = "hazard", title = "Hazard Function (Expression)")

#  Example of plotting an unimodal hazard function using data points

x_data <- seq(0.0000001, 1, by=0.0001)
ecp_plot(data_type = "data", x = x_data, lambda = 2, gamma = 0.3, phi = 30,
func_type = "hazard", title = "Hazard Function (Data Points)")
```

---

ecp_shape   *Bowley skewness and Moors kurtosis*

Description

Computes robust skewness and kurtosis measures based on quantile function of the extended Chen-Poisson (ecp) distribution.

Usage

```
ecp_shape(lambda, gamma, phi, measure)
```

Arguments

lambda, gamma	parameter values > 0.
phi	parameter value != 0.
measure	Specifies the type of measure to be computed. Possible types are:
	- 'bowley' for Bowley skewness, - 'moors' for Moors kurtosis.

Details

The Extended Chen-Poisson distribution has no closed-form expression for the moments. Therefore, the classical measures of skewness and kurtosis based on moments are intractable. In such cases, quantile-based measures are often considered, namely the Bowley skewness and Moors kurtosis.

Value

Numeric value of the Bowley skewness or the Moors kurtosis.

Examples

```
ecp_shape(2, 0.3, 30, measure = 'bowley') # Bowley skewness
```

---

hecp   *Hazard function*

Description

Compute the hazard function of the extended Chen-Poisson (ecp) distribution.

Usage

```
hecp(x, lambda, gamma, phi, log = FALSE)
```

Arguments

x	vector of quantiles.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.
log	logical value

Value

Numeric value of the hazard function.

If log = TRUE, numeric value of the logarithm of the function.

Examples

```
hecp(2, 1, 1, 1, log = FALSE) # hazard function
```

---

pecp   *Distribution function*

Description

Compute the cumulative distribution function of the extended Chen-Poisson (ecp) distribution.

Usage

```
pecp(q, lambda, gamma, phi, lower_tail = TRUE, log_p = FALSE)
```

Arguments

q   vector of quantiles.

lambda, gamma   parameter values > 0.

phi   parameter value != 0.

lower_tail   similar to lower.tail

log_p   logical value

Value

Numeric value of the distribution function.

If lower_tail = FALSE, numeric value of the survival function.

If log_p = TRUE, numeric value of the logarithm of the function.

Examples

```
pecp(2, 1, 1, 1, lower_tail = TRUE, log_p = FALSE) # distribution function
pecp(2, 1, 1, 1, lower_tail = FALSE, log_p = FALSE) # survival function
```

---

qecp   *Quantile function*

Description

Compute the quantile function of the extended Chen-Poisson (ecp) distribution.

Usage

```
qecp(p, lambda, gamma, phi, lower_tail = TRUE, log_p = FALSE)
```

Arguments

p   vector of probabilities.

lambda, gamma   parameter values > 0.

phi   parameter value != 0.

lower_tail   similar to lower.tail.

log_p   logical value.

Value

Numeric value of the quantile function.

If lower_tail = FALSE, numeric value of the logarithm of the function.

If log_p = TRUE, numeric value of the logarithm of the function.

Examples

```
qecp(0.5, 2, 1, 1, lower_tail = TRUE, log_p = FALSE)
# quantile function
```

---

recp   *Function to generate a pseudo-random sample*

Description

Generate a pseudo-random sample, without censoring, from the extended Chen-Poisson (ecp) distribution.

Usage

```
recp(n, lambda, gamma, phi)
```

Arguments

n	sample size.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.

Value

A vector of randomly generated numbers from the extended Chen-Poisson distribution.

Examples

```
recp(10,1,1,1) # random sample of size 10
```

---

secp   *Survival function*

Description

Compute the survival function of the extended Chen-Poisson (ecp) distribution.

Usage

```
secp(q, lambda, gamma, phi, lower_tail = FALSE, cum_haz = FALSE)
```

Arguments

q	vector of quantiles.
lambda, gamma	parameter values > 0.
phi	parameter value != 0.
lower_tail	similar to lower.tail
cum_haz	logical value

Value

Numeric value of the survival function.

If lower_tail = TRUE, numeric value of the distribution function.

If cum_haz = TRUE, numeric value of the cumulative hazard function.

Examples

```
secp(2, 1, 1, 1, lower_tail = FALSE, cum_haz = FALSE) # survival function
secp(2, 1, 1, 1, lower_tail = TRUE, cum_haz = FALSE) # distribution function
secp(2, 1, 1, 1, lower_tail = FALSE, cum_haz = TRUE) # cumulative
# hazard function
```

Index