Package ‘epandist’

August 29, 2016

Title Statistical Functions for the Censored and Uncensored Epanechnikov Distribution

Version 1.1.1

Description Analyzing censored variables usually requires the use of optimization algorithms. This package provides an alternative algebraic approach to the task of determining the expected value of a random censored variable with a known censoring point. Likewise this approach allows for the determination of the censoring point if the expected value is known. These results are derived under the assumption that the variable follows an Epanechnikov kernel distribution with known mean and range prior to censoring. Statistical functions related to the uncensored Epanechnikov distribution are also provided by this package.

Depends R (>= 3.0.0)

License LGPL

LazyData true

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Mathias Borritz Milfeldt [aut, cre]

Maintainer Mathias Borritz Milfeldt <mathias@milfeldt.dk>

Repository CRAN

Date/Publication 2016-02-04 16:43:37

R topics documented:

cepan *Calculate censoring point*

Description

This function calculates the censoring point of a random censored epanechnikov-distributed variable associated a given expected value. The inverse of this function is `evepan`.

Usage

```
cepan(ev, mu = 0, r = 5^0.5, side_censored = "left")
```

Arguments

`ev`	expected value.
`mu`	mean of distribution prior to censoring.
`r`	half the range of the distribution, ie the distance from the mean to the smallest/largest value supported by the distribution. `r=5^.5` corresponds to a standard deviation of 1.
`side_censored`	indicates whether the variable is `left` or `right` censored. Default is `side_censored='left'`

Value

the censoring point associated with `ev`, `mu` and `r`.

Examples

```
#Censoring point of a left-censored epan-distributed variable
#with an expected value of 3 (given mu=0 and r=16):
cepan(ev=3,mu=0,r=16)

#Censoring point of a right-censored epan-distributed variable
#with an expected value of 103 (given mu=100 and r=32):
cepan(ev=94,mu=100,r=32,side_censored="right")
#Results are usually not an integer though and rarely coinciding with mu
```

depan *Probability density function (pdf) for an uncensored epanechnikov distribution*

Description

This function is simply a polynomial of second degree.

Usage

```
depan(x = 0, mu = 0, r = 5^0.5)
```

Arguments

`x`	point on x-axis.
`mu`	mean of distribution.
`r`	half the range of the distribution, ie the distance from the mean to the smallest/largest value supported by the distribution. `r=5^.5` corresponds to a standard deviation of 1.

Value

point density associated with `x`, `mu` and `r`.

Examples

```
#Probability distribution function, epanechnikov:
curve(depan(x),col='blue',ylim=c(0,.4),xlim=c(-3.5,3.5),yaxs='i',xaxs='i',
main='Probability distribution function',ylab='Probability')

#Probability distribution function, normal:
curve(dnorm(x),col='green',add=TRUE)

#Legend
legend(x=-3.5,y=.4,legend=c('Epanechnikov pdf','Normal pdf'),lty=c(1,1),col=c('blue','green'))
```

---

evepan — *Calculate expected value of censored variable*

Description

This function calculates the expected value of a random censored epanechnikov-distributed variable with a given censoring point. The inverse of this function is `cepan`.

Usage

```
evepan(c = 0, mu = 0, r = 5^0.5, side_censored = "left")
```

Arguments

`c`	censoring point.
`mu`	mean of distribution prior to censoring.
`r`	half the range of the distribution, ie the distance from the mean to the smallest/largest value supported by the distribution. `r=5^.5` corresponds to a standard deviation of 1.
`side_censored`	indicates whether the variable is `left` or `right` censored. Default is `side_censored='left'`

Value

the expected value associated with `c`, `mu` and `r`.

Examples

```
#Expected value of an epan-distributed variable left-censored at 100 (given mu=100 and r=10):
evepan(c=100,mu=100,r=10)

#Expected value as a function of censoring point, epanechnikov distribution:
curve(evepan(c=x),col='blue',xlim=c(-sqrt(5),sqrt(5)),yaxs='i',xaxs='i',
main='Expected value as a function of censoring point',xlab='Censoring point',ylab='Expected value')

#Expected value as a function of censoring point, normal distribution:
curve(dnorm(x)+pnorm(x)*x,col='green',add=TRUE)

#Expected value as a function of censoring point, no uncertainty:
curve(1*x,col='grey',add=TRUE)

#Legend
legend(x=-sqrt(5),y=sqrt(5),legend=c('Epanechnikov','Normal distribution','No uncertainty'),
lty=c(1,1),col=c('blue','green','grey'))
```

---

pepan — *Cumulative distribution function (cdf) for an uncensored epanechnikov distribution*

Description

The inverse of this function is qepan.

Usage

```
pepan(x = 0, mu = 0, r = 5^0.5)
```

Arguments

`x`	point on x-axis.
`mu`	mean of distribution.
`r`	half the range of the distribution, ie the distance from the mean to the smallest/largest value supported by the distribution. `r=5^.5` corresponds to a standard deviation of 1.

Value

probability of value below `x` given `mu` and `r`.

Examples

```
#Probability of a value below -1.96:
pepan(x=-1.96,mu=0,r=5^.5)

#Cumulative distribution function of epanechnikov distribution:
curve(pepan(x),col='blue',xlim=c(-2.5,2.5),yaxs='i',xaxs='i',
main='Cumulative distribution function',ylab='Probability')

#Cumulative distribution function of standard normal distribution:
curve(pnorm(x),col='green',add=TRUE)

#Legend
legend(x=-2.5,y=1,legend=c('Epanechnikov cdf','Normal cdf'),lty=c(1,1),col=c('blue','green'))
```

---

qepan *Quantile function for an uncensored epanechnikov distribution*

Description

The inverse of this function is pepan.

Usage

```
qepan(p, mu = 0, r = 5^0.5)
```

Arguments

p	probability.
mu	mean of distribution.
r	half the range of the distribution, ie the distance from the mean to the smallest/largest value supported by the distribution. `r=5^.5` corresponds to a standard deviation of 1.

Value

the quantile associated with `x`, `mu` and `r`.

Examples

```
#Calculate the lower quartile of an epan-distributed variable:
qepan(p=.25,mu=0,r=sqrt(5))

#Use qepan to confirm analytical solution
#Find the quantile corresponding to p=(5+sqrt(5))/8=.9045 when mu=0 and r=sqrt(5):
qepan(p=(5+sqrt(5))/8,mu=0,r=sqrt(5))
#This is equal to
(5-sqrt(5))/2
```

`repan` *Generate random uncensored epanechnikov-distributed data*

Description

This function works in conjuncture with `qepan` and `runif`

Usage

```
repan(n, mu = 0, r = 5^0.5)
```

Arguments

`n`	number of data points.
`mu`	mean of distribution.
`r`	half the range of the distribution, ie the distance from the mean to the smallest/largest value supported by the distribution. `r=5^.5` corresponds to a standard deviation of 1.

Value

vector of random variables.

Examples

```
#Generate and plot 10000 random observations:
hist(repan(10000,mu=100,r=10))
```

Index