Package 'DISTRIB'

October 12, 2022

Type Package

Title Four Essential Functions for Statistical Distributions Analysis: A New Functional Approach

Version 1.0

Date 2016-12-23

Author Abbas Parchami (Department of Statistics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran)

Maintainer Abbas Parchami <parchami@uk.ac.ir>

Description A different way for calculating pdf/pmf, cdf, quantile and random data such that the user is able to consider the name of related distribution as an argument and so easily can changed by a changing argument by user. It must be mentioned that the core and computation base of package 'DISTRIB' is package 'stats'. Although similar functions are introduced previously in package 'stats', but the package 'DISTRIB' has some special applications in some special computational programs.

License LGPL (>= 3)

NeedsCompilation no

Repository CRAN

Date/Publication 2016-12-26 10:34:17

R topics documented:

DISTRIB-package — *Four Essential Functions for Statistical Distributions Analysis: A New Functional Approach*

Description

Previously, four useful functions dnorm, pnorm, qnorm and rnorm are introduced for any univariate distribution in package stats. But the name of these functions are different for any distribution, for example compare the names of dcauchy and dchisq which are for Cauchy and Chi-square distributions, respectively. Now suppose that you have a fixed formula which you want to work with any distribution. To this matter, the author of this package try to define four functions pdf, cdf, rd and q in package DISTRIB, in which the user is able to consider the name of distribution as a argument of them, and therefore these functions can work with any arbitrary distribution.

Author(s)

Abbas Parchami

Maintainer: Abbas Parchami <parchami@uk.ac.ir>

Examples

```
# An applied example for computing p-value in testing H0: mu>=0, vs, H1: mu<0 with two
# different test statistic distribution:

# (a) where the statistics test is T~N(0,1) and its observed value is t = -1.5
p_value = cdf(T.dist="norm", T.dist.par=c(0,1), t=-1.5)
print(p_value)

# (b) where the statistics test T has t-student dist. with 10 degree of freedom
#     and its observed value is t = -1.5
p_value = cdf(T.dist="t", T.dist.par=10, t=-1.5)
print(p_value)
```

cdf — *Cumulative density function (cdf)*

Description

This function compute the Cumulative Density Function (cdf) value of any univariate distribution in point t, i.e. $P(T \leq t)$. Unlike the common cdf's of other distributions (such as pnorm, ppois and etc.) the name of the introduced cdf function is fix in which the name of distribution is considered as an argument. So the cdf function is applicable for any kind of distribution with a unique form but by considering the name of distribution as a parameter (argument) of cdf function.

Usage

```
cdf(T.dist, T.dist.par, t)
```

Arguments

`T.dist`	The distribution name of the random variable is determined by characteristic element `T.dist`. The names of distributions is similar to `stats` package.
`T.dist.par`	A vector of distribution parameters with considered ordering in `stats` package.
`t`	A real number or a vector of real numbers. `cdf` function compute the cumulative density function (cdf) of a distribution in point `t`.

Value

This function gives the value of cumulative density function (cdf) at point `t`.

Examples

```
# Example:
cdf(T.dist="norm", T.dist.par=c(0,1), 0)
cdf(T.dist="t", T.dist.par=c(7), -2)
cdf(T.dist="pois", T.dist.par=5, 0) # Equal to  dpois(0,5)
cdf(T.dist="pois", T.dist.par=5, 5)


## The function is currently defined as
function (T.dist, T.dist.par, t)
{
    pDis = paste("p", T.dist, sep = "", collapse = "")
    if (length(T.dist.par) == 1) {
        cdf.t = do.call(pDis, list(t, T.dist.par[1]))
    }
    else {
        if (length(T.dist.par) == 2) {
            cdf.t = do.call(pDis, list(t, T.dist.par[1], T.dist.par[2]))
        }
        else {
            cdf.t = do.call(pDis, list(t, T.dist.par[1], T.dist.par[2],
                T.dist.par[3]))
        }
    }
    return(cdf.t)
  }
```

`pdf` *Probability density function (pdf) and probability mass function (pmf)*

Description

This function compute the value of Probability Density/Mass Function (pdf/pmf) for any univariate distribution at point t, i.e. $f(t)$ for continues random variable T, or $P(T = t)$ for discrete random variable. Unlike the common pdf's/pmf's of other distributions (such as `dnorm`, `dpois` and etc.) the name of the introduced `pdf` function is fix for any distribution and the name of distribution is considered as an argument of this function. So the `pdf` function is applicable for any kind of distribution with an unique form but by considering the name of T distribution (and its parameters) as two arguments of `pdf` function.

Usage

```
pdf(T.dist, T.dist.par, t)
```

Arguments

`T.dist`	The distribution name of the random variable is determined by characteristic element `T.dist`. The names of distributions is similar to `stats` package.
`T.dist.par`	A vector of distribution parameters with considered ordering in `stats` package.
`t`	A real number or a vector of real numbers. `pdf` compute pdf/pmf of a distribution in point `t`.

Value

This function gives the value of probability density function (pdf) at point `t` for continues random variable, or gives the value of probability mass function (pmf) at point `t` for discrete random variable.

Examples

```
pdf(T.dist="norm", T.dist.par=c(0,1), t=0) # Is equal to  dnorm(0)
pdf(T.dist="t", T.dist.par=c(7), -2) # Is equal to  dt(-2,7)
pdf(T.dist="pois", T.dist.par=5, 5) # Is equal to  dpois(5,5)


## The function is currently defined as
function (T.dist, T.dist.par, t)
{
    dDis = paste("d", T.dist, sep = "", collapse = "")
    if (length(T.dist.par) == 1) {
        pdf.t = do.call(dDis, list(t, T.dist.par[1]))
    }
    else {
        if (length(T.dist.par) == 2) {
            pdf.t = do.call(dDis, list(t, T.dist.par[1], T.dist.par[2]))
        }
        else {
            pdf.t = do.call(dDis, list(t, T.dist.par[1], T.dist.par[2],
                T.dist.par[3]))
        }
    }
    return(pdf.t)
  }
```

q — *Quantile of a distribution*

Description

This function computes the p-th quantile for any common univariate distribution, s.t. $0 \leq p \leq 1$. Unlike the usual quantile functions of other distributions (such as `qnorm`, `qpois` and etc.) the name of the introduced quantile function is fix for any distribution and the name of corresponded distribution is considered as an argument in this function. Thus `q` function is applicable for any kind of distribution with a unique form but by considering the name of corresponded distribution and its parameters as two arguments of `q` function.

Usage

```
q(p, T.dist, T.dist.par)
```

Arguments

`p`	A numeric vector (or single real number) of probabilities in bound [0,1]. Function `q` compute the p-th quantile for the introduced distribution in its arguments part.
`T.dist`	The distribution name of the random variable is determined by characteristic element `T.dist`. The names of distributions is similar to `stats` package.
`T.dist.par`	A vector of distribution parameters with considered ordering in `stats` package.

Value

This function gives the p-th quantile of a given distribution for the real-valued or vector-valued $p \in [0, 1]$.

Examples

```
q(p=0.25, T.dist="norm", T.dist.par=c(0,1)) # Or the first Quartile of N(0,1), i.e. qnorm(0.25)
q(p=1, T.dist="norm", T.dist.par=c(10,2)) # Is equal to  qnorm(1,10,2)
q(0.9, T.dist="cauchy", T.dist.par=c(3,1)) # Is equal to the 9th Decile of Cauchy
                     # distribution with parameters (3,1); i.e.  qcauchy(0.5,3,1)
q(0.237, T.dist="pois", T.dist.par=25) # Is equal to  qpois(0.237,25)


## The function is currently defined as
function (p, T.dist, T.dist.par)
{
    qDis = paste("q", T.dist, sep = "", collapse = "")
    if (length(T.dist.par) == 1) {
        q.t = do.call(qDis, list(p, T.dist.par[1]))
    }
    else {
        if (length(T.dist.par) == 2) {
```

```
            q.t = do.call(qDis, list(p, T.dist.par[1], T.dist.par[2]))
        }
        else {
            q.t = do.call(qDis, list(p, T.dist.par[1], T.dist.par[2],
                T.dist.par[3]))
        }
    }
    return(q.t)
  }
```

`rd` *Compute random data from a distribution*

Description

This function creates random data from any usual univariate distribution in `R`. Unlike the common methods for creating random data from a distribution (such as `rnorm`, `rpois` and etc.), the name of the introduced function is fix for any distribution and the name of distribution is considered as an argument in `rd` function. Therefore creating random data by `rd` function is applicable for any kind of distribution with an unique function form but by considering the name of T distribution (and its parameters) as two arguments of `rd` function.

Usage

```
rd(n, T.dist, T.dist.par)
```

Arguments

`n`	The number of random observations which must be created from the considered distribution in argument part.
`T.dist`	The distribution name of the random variable is determined by characteristic element `T.dist`. The names of distributions is similar to `stats` package.
`T.dist.par`	A vector of distribution parameters with considered ordering in `stats` package.

Value

This function will create `n` random data from the considered distribution.

Examples

```
rd(n=10, T.dist="norm", T.dist.par=c(0,1)) # Is equal to  rnorm(10)
rd(25, T.dist="pois", T.dist.par=3.3) # Is equal to  rpois(25,3.3)

## The function is currently defined as
function (n, T.dist, T.dist.par)
{
    rDis = paste("r", T.dist, sep = "", collapse = "")
    if (length(T.dist.par) == 1) {
```

```
        rd.t = do.call(rDis, list(n, T.dist.par[1]))
    }
    else {
        if (length(T.dist.par) == 2) {
            rd.t = do.call(rDis, list(n, T.dist.par[1], T.dist.par[2]))
        }
        else {
            rd.t = do.call(rDis, list(n, T.dist.par[1], T.dist.par[2],
                T.dist.par[3]))
        }
    }
    return(rd.t)
}
```

Index