Package: ppcc (via r-universe)

October 18, 2024

Type Package

Version 1.2

Date 2020-02-01

Title Probability Plot Correlation Coefficient Test

Author Thorsten Pohlert

Maintainer Thorsten Pohlert <thorsten.pohlert@gmx.de>

Description Calculates the Probability Plot Correlation Coefficient
(PPCC) between a continuous variable X and a specified
distribution. The corresponding composite hypothesis test that
was first introduced by Filliben (1975) <doi:10.1080/00401706.1975.10489279> can be performed to test
whether the sample X is element of either the Normal,
log-Normal, Exponential, Uniform, Cauchy, Logistic, Generalized
Logistic, Gumbel (GEVI), Weibull, Generalized Extreme Value,
Pearson III (Gamma 2), Mielke's Kappa, Rayleigh or Generalized
Logistic Distribution. The PPCC test is performed with a fast
Monte-Carlo simulation.

Depends R(>= 3.0.0)

Suggests VGAM (>= 1.0), nortest(>= 1.0)

License GPL-3

NeedsCompilation yes

RoxygenNote 7.0.2

Classification/MSC-2010 62G10

Repository CRAN

Date/Publication 2020-02-01 20:50:02 UTC

Contents

ppcc-package	*Goodness-of-Fit Tests using the Probability Plot Correlation Coefficient*

Description

The function ppccTest performs the Probability Plot Correlation Coefficient test for various continuous distribution functions.

ppccTest	*Probability Plot Correlation Coefficient Test*

Description

Performs the Probability Plot Correlation Coeffient Test of Goodness-of-Fit

Usage

```
ppccTest(
  x,
  qfn = c("qnorm", "qlnorm", "qunif", "qexp", "qcauchy", "qlogis", "qgumbel",
    "qweibull", "qpearson3", "qgev", "qkappa2", "qrayleigh", "qglogis"),
  shape = NULL,
  ppos = NULL,
  mc = 10000,
  ...
)
```

Arguments

x	a numeric vector of data values; NA values will be silently ignored.
qfn	a character vector naming a valid quantile function
shape	numeric, the shape parameter for the relevant distribution, if applicable; defaults to NULL
ppos	character, the method for estimating plotting point positions, default's to NULL, see Details for corresponding defaults and ppPositions for available methods
mc	numeric, the number of Monte-Carlo replications, defaults to 10000
...	further arguments, currently ignored

Details

Filliben (1975) suggested a probability plot correlation coeffient test to test a sample for normality. The ppcc is defined as the product moment correlation coefficient between the ordered data $x_{(i)}$ and the order statistic medians M_i,

$$r = \frac{\sum_{i=1}^n \left(x_{(i)} - \bar{x}\right) \, \left(M_i - \bar{M}\right)}{\sqrt{\sum_{i=1}^n \left(x_{(i)} - \bar{x}\right)^2 \, \sum_{j=1}^n \left(M_j - \bar{M}\right)^2}},$$

whereas the ordered statistic medians are related to the quantile function of the standard normal distribution, $M_i = \phi^{-1}(m_i)$. The values of m_i are estimated by plotting-point position procedures (see `ppPositions`).

In this function the test is performed by Monte-Carlo simulation:

1. Calculate quantile-quantile $\hat{r}$ for the ordered sample data `x` and the specified `qfn` distribution (with `shape`, if applicable) and given `ppos`.
2. Draw `n` (pseudo) random deviates from the specified `qfn` distribution, where `n` is the sample size of `x`.
3. Calculate quantile-quantile r_i for the random deviates and the specified `qfn` distribution with given `ppos`.
4. Repeat step 2 and 3 for $i = \{1, 2, \ldots, mc\}$.
5. Calculate $S = \sum_{i=1}^n \mathrm{sgn}(\hat{r} - r_i)$ with sgn the sign-function.
6. The estimated p-value is $p = S/mc$.

The probability plot correlation coeffient is invariant for `location` and `scale`. Therefore, the null hypothesis is a composite hypothesis, e.g. $H0 : X \in N(\mu, \sigma), \;\; \mu \in R, \;\; \sigma \in R_{>0}$. Furthermore, distributions with one (additional) specified `shape` parameter can be tested.

The magnitude of $\hat{r}$ depends on the selected method for plotting-point positions (see `ppPositions`) and the sample size. Several authors extended Filliben's method to assess the goodness-of-fit to other distributions, whereas theoretical quantiles were used as opposed to Filliben's medians.

The default plotting positions (see `ppPositions`) depend on the selected `qfn`.

Distributions with none or one single scale parameter that can be tested:

Argument	Function	Default pppos	Reference
`qunif`	Uniform	`Weibull`	Vogel and Kroll (1989)
`qexp`	Exponential	`Gringorton`	
`qgumbel`	Gumbel	`Gringorton`	Vogel (1986)
`qrayleigh`	Rayleigh	`Gringorton`	

Distributions with `location` and `scale` parameters that can be tested:

Argument	Function	Default pppos	Reference
`qnorm`	Normal	`Blom`	Looney and Gulledge (1985)
`qlnorm`	log-Normal	`Blom`	Vogel and Kroll (1989)
`qcauchy`	Cauchy	`Gringorton`	

qlogis	Logistic	Blom

If Blom's plotting position is used for qnorm, than the ppcc-test is related to the Shapiro-Francia normality test (Royston 1993), where $W' = r^2$. See sf.test and example(ppccTest).

Distributions with additional shape parameters that can be tested:

Argument	Function	Default pppos	Reference
qweibull	Weibull	Gringorton	
qpearson3	Pearson III	Blom	Vogel and McMartin (1991)
qgev	GEV	Cunane	Chowdhury et al. (1991)
qkappa2	two-param. Kappa Dist.	Gringorton	
qglogis	Generalized Logistic	Gringorton	

If qfn = qpearson3 and shape = 0 is selected, the qnorm distribution is used. If qfn = qgev and shape = 0, the qgumbel distribution is used. If qfn = qglogis and shape = 0 is selected, the qglogis distribution is used.

Value

a list with class 'htest'

Note

As the pvalue is estimated through a Monte-Carlo simulation, the results depend on the selected seed (see set.seed) and the total number of replicates (mc).

The default of mc = 10000 re-runs is sufficient for testing the composite hypothesis on levels of $\alpha = [0.1, 0.05]$. If a level of $\alpha = 0.01$ is desired, than larger sizes of re-runs (e.g. mc = 100000) might be required.

References

J. U. Chowdhury, J. R. Stedinger, L.-H. Lu (1991), Goodness-of-Fit Tests for Regional Generalized Extreme Value Flood Distributions, *Water Resources Research* 27, 1765–1776.

J. J. Filliben (1975), The Probability Plot Correlation Coefficient Test for Normality, *Technometrics* 17, 111–117.

S. Kim, H. Shin, T. Kim, J.-H. Heo (2010), Derivation of the Probability Plot Correlation Coefficient Test Statistics for the Generalized Logistic Distribution. Intern. Workshop Adv. in Stat. Hydrol., May 23 - 25, 2010 Taormina.

S. W. Looney, T. R. Gulledge (1985), Use of Correlation Coefficient with Normal Probability Plots, *The American Statistician* 39, 75–79.

P. W. Mielke (1973), Another family of distributions for describing and analyzing precipitation data. *Journal of Applied Meteorology* 12, 275–280.

P. Royston, P. (1993), A pocket-calculator algorithm for the Shapiro-Francia test for non-normality: an application to medicine. *Statistics in Medicine* 12, 181-184.

R. M. Vogel (1986), The Probability Plot Correlation Coefficient Test for the Normal, Lognormal, and Gumbel Distributional Hypotheses, *Water Resources Research* 22, 587–590.

R. M. Vogel, C. N. Kroll (1989), Low-flow frequency analysis using probability-plot correlation coefficients, *Journal of Water Resources Planning and Management* 115, 338–357.

R. M. Vogel, D. E. McMartin (1991), Probability Plot Goodness-of-Fit and Skewness Estimation Procedures for the Pearson Type 3 Distribution, *Water Resources Research* 27, 3149–3158.

See Also

qqplot, qqnorm, ppoints, ppPositions, Normal, Lognormal, Uniform, Exponential, Cauchy, Logistic, qgumbel, Weibull, qgev.

Examples

```
## Filliben (1975, p.116)
## Note: Filliben's result was 0.98538
## decimal accuracy in 1975 is assumed to be less than in 2017
x <- c(6, 1, -4, 8, -2, 5, 0)
set.seed(100)
ppccTest(x, "qnorm", ppos="Filliben")
## p between .75 and .9
## see Table 1 of Filliben (1975, p.113)
##
set.seed(100)
## Note: default plotting position for
## qnorm is ppos ="Blom"
ppccTest(x, "qnorm")
## p between .75 and .9
## see Table 2 of Looney and Gulledge (1985, p.78)
##
##
set.seed(300)
x <- rnorm(30)
qn <- ppccTest(x, "qnorm")
qn
## p between .5 and .75
## see Table 2 for n = 30 of Looney and Gulledge (1985, p.78)
##
## Compare with Shapiro-Francia test
if(require(nortest)){
   sn <- sf.test(x)
   print(sn)
   W <- sn$statistic
   rr <- qn$statistic^2
   names(W) <- NULL
   names(rr) <- NULL
   print(all.equal(W, rr))
}
ppccTest(x, "qunif")
ppccTest(x, "qlnorm")
old <- par()
```

```
par(mfrow=c(1,3))
xlab <- "Theoretical Quantiles"
ylab <- "Empirical Quantiles"
qqplot(x = qnorm(ppPositions(30, "Blom")),
       y = x, xlab=xlab, ylab=ylab, main = "Normal q-q-plot")
qqplot(x = qunif(ppPositions(30, "Weibull")),
       y = x, xlab=xlab, ylab=ylab, main = "Uniform q-q-plot")
qqplot(x = qlnorm(ppPositions(30, "Blom")),
       y = x, xlab=xlab, ylab=ylab, main = "log-Normal q-q-plot")
par(old)
##
if (require(VGAM)){
set.seed(300)
x <- rgumbel(30)
gu <- ppccTest(x, "qgumbel")
print(gu)
1000 * (1 -  gu$statistic)
}
##
## see Table 2 for n = 30 of Vogel (1986, p.589)
## for n = 30 and Si = 0.5, the critical value is 16.9
##
set.seed(200)
x <- runif(30)
un <- ppccTest(x, "qunif")
print(un)
1000 * (1 - un$statistic)
##
## see Table 1 for n = 30 of Vogel and Kroll (1989, p.343)
## for n = 30 and Si = 0.5, the critical value is 10.5
##
set.seed(200)
x <- rweibull(30, shape = 2.5)
ppccTest(x, "qweibull", shape=2.5)
ppccTest(x, "qweibull", shape=1.5)
##
if (require(VGAM)){
set.seed(200)
x <- rgev(30, shape = -0.2)
ev <- ppccTest(x, "qgev", shape=-0.2)
print(ev)
1000 * (1 - ev$statistic)
##
## see Table 3 for n = 30 and shape = -0.2
## of Chowdhury et al. (1991, p.1770)
## The tabulated critical value is 80.
}
```

---

ppPositions *Plotting Point Positions*

Description

Calculates plotting point positions according to different authors

Usage

```
ppPositions(
  n,
  method = c("Gringorton", "Cunane", "Filliben", "Blom", "Weibull", "ppoints")
)
```

Arguments

`n`	numeric, the sample size
`method`	a character string naming a valid method (see Details)

Details

The following methods can by selected:

`"Gringorton"` the plotting point positions are calculated as

$$m_i = (i - 0.44) \, / \, (n + 0.12)$$

`"Cunane"` the plotting point positions are calculated as

$$m_i = (i - 0.4) \, / \, (n + 0.2)$$

`"Blom"` the plotting point positions are calculated as

$$m_i = (i - 0.3175) \, / \, (n + 0.25)$$

`"Filliben"` the order statistic medians are calculated as:

$$m_i = \begin{cases} 1 - 0.5^{1/n} & i = 1 \\ (i - 0.3175) \, / \, (n + 0.365) & i = 2, \ldots, n-1 \\ 0.5^{1/n} & i = n \end{cases}$$

`"ppoints"` R core's default plotting point positions are calculated (see `ppoints`).

Value

a vector of class numeric that contains the plotting positions

Index