Package ‘qqtest’

Type Package
Title Quantile Quantile Plots Self Calibrating For Visual Testing
Version 1.0
Date 2014-12-02
Maintainer Wayne Oldford <rwoldford@uwaterloo.ca>
Description Provides the function qqtest which incorporates uncertainty in its qqplot display(s) so that the user might have a better sense of the evidence against the specified distributional hypothesis. qqtest draws a quantile quantile plot for visually assessing whether the data come from a test distribution that has been defined in one of many ways. The vertical axis plots the data quantiles, the horizontal those of a test distribution. The default behaviour generates 1000 samples from the test distribution and overlays the plot with pointwise interval estimates for the ordered quantiles from the test distribution. A small number of independently generated exemplar quantile plots are also overlaid. Both the interval estimates and the exemplars provide different comparative information to assess the evidence provided by the qqplot for or against the hypothesis that the data come from the test distribution. Finally, a visual test of significance (a lineup plot) can also be displayed to test the null hypothesis that the data come from the test distribution.
License GPL-3
Depends R (>= 2.10.0)
Imports grDevices, MASS
Author Wayne Oldford [aut, cre]
NeedsCompilation no
Repository CRAN
Date/Publication 2014-12-14 01:15:35

R topics documented:

   dKay ........................................................... 2
The density function of the K distribution

Description

The K density function on df degrees of freedom and non-centrality parameter ncp.

A K distribution is the square root of a chi-square divided by its degrees of freedom. That is, if x is chi-squared on m degrees of freedom, then y = sqrt(x/m) is K on m degrees of freedom. Under standard normal theory, K is the distribution of the pivotal quantity s/sigma where s is the sample standard deviation and sigma is the standard deviation parameter of the normal density. K is the natural distribution for tests and confidence intervals about sigma. K densities are more nearly symmetric than are chi-squared and concentrate near 1. As the degrees of freedom increase, they become more symmetric, more concentrated, and more nearly normally distributed.

Usage

dkay(x, df, ncp = 0, log.p = FALSE)

Arguments

x A vector of values at which to calculate the density.
df Degrees of freedom (non-negative, but can be non-integer).
ncp Non-centrality parameter (non-negative).
log.p logical; if TRUE, probabilities are given as log(p).

Value

dkay gives the density evaluated at the values of x.

Invalid arguments will result in return value NaN, with a warning.

The length of the result is the maximum of the lengths of the numerical arguments for the other functions.

The numerical arguments are recycled to the length of the result. Only the first elements of the logical arguments are used.

Note

All calls depend on analogous calls to chi-squared functions. See dchisq for details on non-centrality parameter calculations.
Examples

dkay(1, 20)
#
# compare K density to that of chi as degrees of freedom increase
op <- par(mfrow=c(1,2))
p <- seq(0.001, .999, 0.001)
#
# First get all the chi-square densities and plot them
xchi5 <- qchisq(p, 5)
dchi5 <- dchisq(xchi5, 5)
xchi10 <- qchisq(p, 10)
dchi10 <- dchisq(xchi10, 10)
xchi20 <- qchisq(p, 20)
dchi20 <- dchisq(xchi20, 20)
xchi30 <- qchisq(p, 30)
dchi30 <- dchisq(xchi30, 30)
xlim <- range(xchi5, xchi10, xchi20, xchi30)
ylim <- range(dchi5, dchi10, dchi20, dchi30)
plot(xchi5, dchi5, type="l", xlab="x", ylab="density",
    xlim=xlim, ylim=ylim,
    main="chi-squared densities")
lines(xchi10, dchi10, lty=2)
lines(xchi20, dchi20, lty=3)
lines(xchi30, dchi30, lty=4)
legend("topright",
    legend=c("df = 5", "df = 10", "df = 20", "df = 30"),
    lty=c(1,2,3,4),
    title="degrees of freedom",
    cex=0.75, bty="n")
#
# Now get all the K densities and plot them
xkay5 <- qkay(p, 5)
dkay5 <- dkay(xkay5, 5)
xkay10 <- qkay(p, 10)
dkay10 <- dkay(xkay10, 10)
xkay20 <- qkay(p, 20)
dkay20 <- dkay(xkay20, 20)
xkay30 <- qkay(p, 30)
dkay30 <- dkay(xkay30, 30)
xlim <- range(xkay5, xkay10, xkay20, xkay30)
ylim <- range(dkay5, dkay10, dkay20, dkay30)
plot(xkay5, dkay5, type="l",
    xlab="x", ylab="density",
    xlim=xlim, ylim=ylim,
    main="K densities")
lines(xkay10, dkay10, lty=2)
lines(xkay20, dkay20, lty=3)
lines(xkay30, dkay30, lty=4)
legend("topright",
    legend=c("df = 5", "df = 10", "df = 20", "df = 30"),
    lty=c(1,2,3,4),
    title="degrees of freedom")
penicillin

```r
cex=0.75, bty="n")
par(op)
#
# Note how K densities are more nearly symmetric and concentrate near 1.
# As the degrees of freedom increase, they become more symmetric,
# more concentrated, and more nearly normally distributed.
```

### Description

Values are arranged in decreasing order of absolute magnitude. Daniel (1959) uses the data to illustrate the use of half-normal plots. In his words: "We need, of course, some rule of inference that will help us to be objective in judging whether or not the largest effects are real."

### Usage

```r
data("penicillin")
```

### Format

A data frame with 31 rows and 1 variate:

- **value**: a numeric vector containing the value of the contrast

### Details

Name of the contrast effect is given as the row name of each value.

### Source


### References


### Examples

```r
data(penicillin)
## Daniel's half-normal plot (from qqtest package)
qqtest(penicillin, dist="half-normal")
```
The cumulative distribution function for the K distribution on df degrees of freedom having non-centrality parameter ncp.

A K distribution is the square root of a chi-square divided by its degrees of freedom. That is, if x is chi-squared on m degrees of freedom, then y = sqrt(x/m) is K on m degrees of freedom. Under standard normal theory, K is the distribution of the pivotal quantity s/sigma where s is the sample standard deviation and sigma is the standard deviation parameter of the normal density. K is the natural distribution for tests and confidence intervals about sigma. K densities are more nearly symmetric than are chi-squared and concentrate near 1. As the degrees of freedom increase, they become more symmetric, more concentrated, and more nearly normally distributed.

Usage

pkay(q, df, ncp = 0, upper.tail = FALSE, log.p = FALSE)

Arguments

q A vector of quantiles at which to calculate the cumulative distribution.
df Degrees of freedom (non-negative, but can be non-integer).
cp Non-centrality parameter (non-negative).
upper.tail logical; if TRUE, instead of returning F(q) (the default), the upper tail probabilities 1-F(q) = Pr(Q>q) are returned.
log.p logical; if TRUE, probabilities are given as log(p).

Value

pkay returns the value of the K cumulative distribution function, F(q), evaluated at q for the given df and ncp. If upper.tail = TRUE then the upper tail probabilities 1-F(q) = Pr(Q>q) are returned instead of F(q).

Invalid arguments will result in return value NaN, with a warning.

The length of the result is the maximum of the lengths of the numerical arguments.

The numerical arguments are recycled to the length of the result. Only the first elements of the logical arguments are used.

Note

All calls depend on analogous calls to chi-squared functions. See pchisq for details on non-centrality parameter calculations.
Examples

pkay(1, 20)
q <- seq(0.01, 1.8, 0.01)
#
# Plot the cdf for K(5)
u <- pkay(q, 5)
plot(q, u, type="l",
     xlab="q", ylab="cumulative probability",
     xlim=range(q), ylim=c(0,1),
     main="K cdf")
#
# Add some other K cdfs
lines(q, pkay(q,10), lty=2)
lines(q, pkay(q,20), lty=3)
lines(q, pkay(q,30), lty=4)
legend("topleft",
       legend=c("df = 5", "df = 10", "df = 20", "df = 30"),
       lty=c(1,2,3,4),
       title="degrees of freedom",
       cex=0.75, bty="n")

qkay

qkay The K distribution quantile function

Description

Quantile function for the K distribution on df degrees of freedom having non-centrality parameter ncp.

A K distribution is the square root of a chi-square divided by its degrees of freedom. That is, if x is chi-squared on m degrees of freedom, then y = sqrt(x/m) is K on m degrees of freedom. Under standard normal theory, K is the distribution of the pivotal quantity s/sigma where s is the sample standard deviation and sigma is the standard deviation parameter of the normal density. K is the natural distribution for tests and confidence intervals about sigma. K densities are more nearly symmetric than are chi-squared and concentrate near 1. As the degrees of freedom increase, they become more symmetric, more concentrated, and more nearly normally distributed.

Usage

qkay(p, df, ncp = 0, upper.tail = FALSE, log.p = FALSE)

Arguments

p A vector of probabilities at which to calculate the quantiles.
df Degrees of freedom (non-negative, but can be non-integer).
ncp Non-centrality parameter (non-negative).
age_logical; if TRUE, instead of returning F(x) (the default), the upper tail probabil-
log.p logical; if TRUE, probabilities are given as log(p).
qqtest

Value

qkay returns the quantiles at probabilities p for a K on df degrees of freedom and non-centrality parameter ncp.

Invalid arguments will result in return value NaN, with a warning.

The length of the result is the maximum of the lengths of the numerical arguments.

The numerical arguments are recycled to the length of the result. Only the first elements of the logical arguments are used.

Note

All calls depend on analogous calls to chi-squared functions. See qchisq for details on non-centrality parameter calculations.

Examples

p <- ppoints(30)
# Get the quantiles for these points
q5 <- qkay(p, 5)
plot(p, q5, main="Quantile plot of K(20)", ylim=c(0, max(q5)))
# Add quantiles from another K
points(p, qkay(p, 20), pch=19)

#
# Do these EXACT quantiles from a K(5) look like they might
# have been generated from K(20)?
qqtest(q5, dist="kay", df=20)

# How about compared to normal?
qqnorm(q5)
qqtest(q5)
# for this many degrees of freedom it looks a lot like
# a gaussian (normal) distribution

# And should look really good compared to the true distribution
qqtest(q5, dist="kay", df=5)
#
#
# But not so much like it came from a K on 1 degree of freedom
qqtest(q5, dist="kay", df=1)
Description

Draws a quantile quantile plot for visually assessing whether the data come from a test distribution that has been defined in one of many ways. The vertical axis plots the data quantiles, the horizontal those of a test distribution. Interval estimates and exemplars provide different comparative information to assess the evidence provided by the qqplot against the hypothesis that the data come from the test distribution (default is normal or gaussian). Interval estimates provide test information related to individual quantiles, exemplars provide test information related to the shape of the quantile quantile curve. Optionally, a visual test of significance (a lineup plot) can be displayed to provide a coarse level of significance for testing the null hypothesis that the data come from the test distribution. The default behaviour generates 1000 samples from the test distribution and overlays the plot with pointwise interval estimates for the ordered quantiles from the test distribution. A small number of independently generated exemplar test distribution sample quantile traces are also overlaid. Various option choices are available to effect different visualizations of the uncertainty surrounding the quantile quantile plot (see argument descriptions and examples).

Usage

```r
qqtest(data, dist = "gaussian", df = 1, qfunction = NULL, 
rfun = NULL, dataTest = NULL, xAxisAsProbs = TRUE, 
yAxisAsProbs = TRUE, xAxisProbs = c(0.05, 0.25, 0.5, 0.75, 0.95), 
yAxisProbs = c(0.05, 0.25, 0.5, 0.75, 0.95), nreps = 1000, 
centralPercents = c(0.9, 0.95, 0.99), envelope = TRUE, 
drawPercentiles = TRUE, drawQuartiles = TRUE, legend = TRUE, 
nexemplars = 10, plainTrails = FALSE, alphaTrails = 0.5, 
lwdTrails = 1, lineup = FALSE, nsuspects = 20, col = NULL, h = 260, 
c = 90, l = 60, alpha = 1, cex = 1, pch = 19, xlab = NULL, 
ylab = NULL, xlim = NULL, ylim = NULL, ...)
```

Arguments

data A univariate dataset to be tested. If data has more than one column, the first is used.

dist The name of the distribution against which the comparison is made, the test distribution for a few built-in distributions. One of "gaussian" (or "normal"), "log normal", "half normal", "uniform", "student", "chi-squared", or "kay". Only the first three characters of any of these is needed to specify the dist. If dist is "student", "chi-squared", or "kay", then a value for the degrees of freedom argument (df below) is also required.

df Degrees of freedom of dist to be used when dist is either "student" or "chi-squared".

qfunction If non-NULL, this must be a function of a single argument (a proportion, p say) which will be used to calculate the quantiles for the test distribution. If non-NULL, the rfunction should also be non-NULL. The value of the dist argument will be ignored when this is the case.

rfun If non-NULL, this must be a function of a single argument (a count, n say) which will be used to randomly select a sample of size n from the test distribution. If non-NULL, the qfunction must also be non-NULL. If qfunction is non-NULL
and $r$function is NULL, then $q$function will be applied to the output of a call to `runif` in place of the NULL $r$function (i.e., a probability integral transform is used to generate a random sample). The value of the dist argument will be ignored whenever $q$function is a function.

**dataTest**

If non-NULL, this must be a second data set. The empirical distribution given by this data will be used as the test distribution against which the value of data will be tested. If non-NULL, the values of the arguments dist, $q$function, and $r$function will all be ignored in favour of using this empirical distribution as the test distribution.

**xAxisAsProbs**

If TRUE (the default) the horizontal axis will be labelled as probabilities. These are the cumulative probabilities according to the test distribution. They are located at the corresponding quantile values. They are handy in comparing percentiles of the test and data distributions as well as giving some measure of the symmetry and tail weights of the test distribution by their location. If FALSE the axis is labelled according to the quantile values.

**yAxisAsProbs**

If TRUE (the default) the vertical axis will be labelled as probabilities. These are the cumulative probabilities according to the empirical distribution of the data. They are located at the corresponding quantile values. They are handy in comparing percentiles of the test and data distributions as well as giving some measure of the symmetry and tail weights of the data distribution by their location. If FALSE the axis is labelled according to the quantile values.

**xAxisProbs**

A vector of probabilities to be used to label the x axis ticks when xAxisAsProbs is TRUE. Default is c(0.05, 0.25, 0.50, 0.75, 0.95). Ignored if xAxisAsProbs is FALSE.

**yAxisProbs**

A vector of probabilities to be used to label the y axis ticks when yAxisAsProbs is TRUE. Default is c(0.05, 0.25, 0.50, 0.75, 0.95). Ignored if yAxisAsProbs is FALSE.

**nreps**

The number of replicate samples to be taken from the test distribution to construct the pointwise intervals for each quantile. Default is 1000. From these samples, an empirical distribution is generated from the test distribution for the ordered quantiles corresponding to the values of ppoints(length(data)). These are used to construct central intervals of whatever proportions are given by centralPercents.

**centralPercents**

The vector of proportions determining the central intervals of the empirical distribution of each ordered quantile from the test distribution. Default is c(0.90, 0.95, 0.99) corresponding to central 90, 95, and 99% simulated pointwise confidence intervals for each quantile coming from the test distribution for a sample the same size as data. The quality of these interval locations typically increases with nreps and decreases with the probability used for each interval.

**envelope**

If TRUE (the default), a grey envelope is plotted showing the central intervals for each quantile as a shade of grey. The higher is the corresponding probability associated with the interval, the lighter is the shade. The outermost edges of the envelope are the range of the simulated data from the test distribution. The envelope thus provides a (pointwise) density estimate of the quantiles drawn from the test distribution for this sample size. If FALSE no envelope is drawn.
drawPercentiles
If TRUE (the default), a pair of curves is plotted to show each of the central intervals as a different line type. These are plotted over the envelope if envelope is TRUE. If FALSE no simulated percentile curves are drawn.

drawQuartiles
If TRUE (the default), a pair of curves is plotted to show the quartiles (central 50% region) of the ordered quantiles simulated from the test distribution. The median of these is also plotted as a solid line type. These are plotted over the envelope if envelope is TRUE. If FALSE none of these curves are drawn.

legend
If TRUE (the default), a legend for the appearance of the simulated ranges of the central intervals is added to the plot. If FALSE, no legend appears.

nexemplars
The number of replicate samples to be taken from the test distribution and plotted as a coloured trail on the qqplot. Each such trail is a sample of the same size as data but truly coming from the test distribution. Each trail gives some idea of what the shape of a qqplot would be for a sample of that size from the test distribution. Together, they give some sense of the variability in the plot’s shape.

plainTrails
If TRUE, then a single grey colour is used for all exemplar trails. If FALSE (the default), each exemplar trail is shown in a different colour.

alphaTrails
The alpha transparency to be used in plotting all exemplar trails. The default is 0.5. Because the trails will over plot, a lower alphaTrails value is recommended as nexemplars increases.

lwdTrails
The graphical line width (lwd) to be used in plotting all exemplar trails. The default is 1. Because the trails will over plot, combining a larger lwdTrails with envelope = FALSE, a lower alphaTrails value larger nexemplars can give a truer sense of the density of qqplot configurations than with envelope = TRUE.

lineup
If TRUE, the qqplot of data is randomly located in a grid of nsuspects plots. Identical arguments are given to construct all qqtest plots in the grid. Assuming the viewer has not seen the qqplot of this data before, a successful selection of the true data plot out of the grid of plots corresponds to evidence against the hypothesis that the data come from the test distribution. Significance level is 1/nsuspects. Default corresponds to a (1/20) = 5% significance level. Each plot is given a suspect number from 1 to nsuspects (left to right, top to bottom). The suspect number of the plot corresponding to the actual data is returned, slightly obfuscated to help keep the test honest.

nsuspects
The total number of plots to be viewed in the lineup display when lineup is lineup.

col
If non-NULL, col must be colour to be used for the points in the plot. If NULL (the default), an hcl colour will be used from the values of the arguments h, c, l, and alpha.

h
The hue of the colour of the points. Specified as an angle in degrees from 0 to 360 around a colour wheel. E.g. 0 is red, 120 green, 240 blue, Default is 260 (a bluish).

c
The chroma of the colour of the points. Takes values from 0 to an upper bound that is a function of hue, h, and luminance, l. Roughly, for fixed h and l the higher the value of c the greater the intensity of colour.
The luminance of the colour of the points. Takes values from 0 to 100. For any given combination of hue, \( h \), and chroma, \( c \), only a subset of this range will be possible. Roughly, for fixed \( h \) and \( c \) the higher the value of \( l \) the lighter is the colour.

**alpha**

The alpha transparency of the colour of the points. Takes values from 0 to 1. Values near 0 are more transparent, values near 1 are more opaque. Alpha values sum when points over plot, giving some indication of density.

**cex**

The graphical parameter cex for the size of the points.

**pch**

The graphical parameter pch for the point character to be used for the points. Default is 19, a filled circle.

**xlab**

The graphical parameter xlab labelling the x axis of the plot. If NULL (the default), an xlab is created based on the information available from the other arguments to qqtest about the test distribution. An empty string will suppress the labelling.

**ylab**

The graphical parameter ylab labelling the y axis of the plot. If NULL (the default), a ylab is created based on the information available from the other arguments to qqtest. An empty string will suppress the labelling.

**xlim**

The graphical parameter xlim determining the display limits of the x axis.

**ylim**

The graphical parameter ylim determining the display limits of the y axis.

*...*

Any further graphical parameters to be passed to the plot function.

**Value**

Displays the qqplot. If lineup is TRUE, it returns a list with the location (TrueLoc) of the plot that corresponds to data encoded as a string whose contents need to be evaluated. This provides some simple obfuscation of the true location so that the visual assessment can be honest.

**Examples**

```r
# default qqtest plot
qqtest(precip)
#
# default qqtest plot
qqtest(precip, main = "Precipitation (inches/year) in 70 US cities")
#
# compare qqtest default to qqnorm
op <- par(mfrow=c(1,2))
qqnorm(precip); qqtest(precip)
par(op)
#
# gaussian - no quartiles, no exemplars
qqtest(precip, nexemplars=0, drawQuartiles=FALSE, main = "Precipitation (inches/year) in 70 US cities")
#
# gaussian - no quartiles, no percentiles,
# no envelope just coloured exemplars
qqtest(precip, nexemplars=20, envelope=FALSE, drawPercentiles=FALSE, drawQuartiles=FALSE)
```
main = "Precipitation (inches/year) in 70 US cities"
#
# gaussian - no quartiles, no percentiles, no envelope,
# plain grey trails, wide trails show density
qqtest(precip, nreplars=20,
    lwdTrails=10, plainTrails=TRUE, alphaTrails=0.3,
    envelope=FALSE, drawPercentiles=FALSE, drawQuartiles=FALSE,
    main = "Precipitation (inches/year) in 70 US cities")
#
# gaussian - no quartiles, no percentiles, no envelope,
# colour trails, wide trails show density
qqtest(precip, nreplars=20, lwdTrails=10, col="black",
    plainTrails=FALSE, alphaTrails=0.3,
    envelope=FALSE, drawPercentiles=FALSE, drawQuartiles=FALSE,
    main = "Precipitation (inches/year) in 70 US cities")
#
# gaussian - common qqplot like qqnorm
qqtest(precip, xAxisAsProbs=FALSE, yAxisAsProbs=FALSE,
    nreplars=0, envelope=FALSE,
    drawPercentiles=FALSE, drawQuartiles=FALSE,
    col="black", main = "Normal Q-Q Plot",
    xlab="Theoretical Quantiles",
    ylab="Precipitation (inches/year) in 70 US cities",
    pch=21)
#
# gaussian - traditional qqplot, but now showing in the line up
result <- qqtest(precip, nreplars=0, nreps=0,
    envelope=FALSE,
    drawPercentiles=FALSE, drawQuartiles=FALSE,
    lineup=TRUE,
    cex=0.75, col="grey20",
    xlab="", ylab="",
    pch=21)
# the location of the real data in the line up can be found by evaluating
# the contents of the string
result$trueLoc
#
# lognormal
qqtest(precip, dist = "lognormal",
    main = "Precipitation (inches/year) in 70 US cities")
#
#
# Half normal ... using the penicillin data from Daniel(1959)
data(penicillin)
qqtest(penicillin, dist = "half-normal")

# Or the same again but with significant contrast labelled

with(penicillin,
{qqtest(value, xAxisProbs=c(0.1, 0.75, 0.90, 0.95),
    dist="half-normal",}
ppAdj <- (1+ppoints(3))/2  # to get half-normals from normal
x <- qnorm(ppAdj)
valOrder <- order(value)  # need data and rownames in increasing order
y <- value[valOrder]
tag <- rownames(penicillin)[valOrder]
selPoints <- 28:31  # going to label only the largest effects
    text(x[selPoints], y[selPoints],
    tags[selPoints],
    pos=2, cex=0.75)
}
# student on 3 df
qqtest(precip, dist = "student", df = 3,
    main = "Precipitation (inches/year) in 70 US cities")
# chi-squared on 3 df
qqtest(precip, dist = "chi-squared", df = 3,
    main = "Precipitation (inches/year) in 70 US cities")
# user supplied qfunction and rfunction -- compare to beta distribution
    qfunction=function(p){qbeta(p, 2, 2)},
    rfunction=function(n){rbeta(n, 2, 2)},
    main = "Precipitation (inches/year) in 70 US cities")
# user supplied qfunction only -- compare to beta distribution
qqtest(precip,
    qfunction=function(p){qbeta(p, 2, 2)},
    main = "Precipitation (inches/year) in 70 US cities")
# comparing data samples
# Does the sample of beaver2's temperatures look like they
# could have come from a distribution shaped like beaver1's?
    qqtest(beaver2[,"temp"],
        dataTest=beaver1[,"temp"],
        ylab="Beaver 2", xlab="Beaver 1",
        main="Beaver body temperatures")
# For the famous iris data, does the sample of iris versicolor
# appear to have the same (marginal) distributional shape
# as does that of iris virginica (to which it is more closely related)?
    op <- par(mfrow=c(2,2))
    with(iris, {
    qqtest(Sepal.Length[Species=="versicolor"],
        dataTest= Sepal.Length[Species=="virginica"],}
The K distribution - generating pseudo-random values

Description

Random generation for the K distribution on df degrees of freedom having non-centrality parameter ncp.

A K distribution is the square root of a chi-square divided by its degrees of freedom. That is, if x is chi-squared on m degrees of freedom, then y = sqrt(x/m) is K on m degrees of freedom. Under standard normal theory, K is the distribution of the pivotal quantity s/sigma where s is the sample standard deviation and sigma is the standard deviation parameter of the normal density. K is the natural distribution for tests and confidence intervals about sigma. K densities are more nearly symmetric than are chi-squared and concentrate near 1. As the degrees of freedom increase, they become more symmetric, more concentrated, and more nearly normally distributed.

Usage

rkay(n, df, ncp = 0)

Arguments

n Number of observations. If length(n) > 1, the length is taken to be the number required.
df Degrees of freedom (non-negative, but can be non-integer).
cp Non-centrality parameter (non-negative).

Value

rkay returns pseudo-randomly generated values.
Invalid arguments will result in return value NaN, with a warning.
**Note**

Depends on call to analogous chi-squared functions. See `rchisq` for details on non-centrality parameter calculations.

**Examples**

```r
x <- rkay(100, 20)
hist(x, main="100 observations from a K(20)")
# Certainly looks like it comes from a K on 20
qqtest(x, dist="kay", df=20)
# for this many degrees of freedom it looks
# a lot like a gaussian (normal) distribution
qqtest(x, dist="gau", df=1)
# But not like it came from a K on 1 degree of freedom
qqtest(x, dist="kay", df=1)
#```
Index

*Topic datasets
  penicillin, 4

dkay, 2
penicillin, 4
pkay, 5
qkay, 6
qqtest, 7
rkay, 14