

Package ‘Rgof’

March 13, 2025

Title 1d Goodness of Fit Tests

Version 3.1.0

Description Routines that allow the user to run a large number of goodness-of-fit tests.

It allows for data to be continuous or discrete. It includes routines to estimate the power of the tests and display them as a power graph.

The routine `run.studies` allows a user to quickly study the power of a new method and how it compares to some of the standard ones.

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.3.2

LinkingTo Rcpp

Imports Rcpp, parallel, ggplot2, stats, graphics, microbenchmark

Suggests rmarkdown, knitr

VignetteBuilder knitr

Depends R (>= 3.5)

LazyData true

NeedsCompilation yes

Author Wolfgang Rolke [aut, cre] (<<https://orcid.org/0000-0002-3514-726X>>)

Maintainer Wolfgang Rolke <wolfgang.rolke@upr.edu>

Repository CRAN

Date/Publication 2025-03-13 19:10:02 UTC

Contents

case.studies	2
check.functions	3
chi_power_cont	3
chi_power_disc	4
chi_test_cont	5
chi_test_disc	6

gof_power	7
gof_power_cont	9
gof_power_disc	10
gof_test	11
gof_test_adjusted_pvalue	13
gof_test_cont	15
gof_test_cont_adj	16
gof_test_disc	17
gof_test_disc_adj	19
make_bins_cont	20
make_bins_disc	21
newTSdisc	21
plot_power	22
power_cont_R	22
power_disc_R	23
power_newtest	25
power_studies_results	26
pvaluecdf	26
run.studies	27
signif.digits	28
timecheck	29
TS_cont	29
TS_disc	30

Index 31

case.studies	<i>This function creates the functions needed to run the various case studies.</i>
--------------	--

Description

This function creates the functions needed to run the various case studies.

Usage

```
case.studies(which, nsample = 500)
```

Arguments

which	name of the case study.
nsample	=500, sample size.

Value

a list of functions

check.functions	<i>This function checks whether the inputs have the correct format</i>
-----------------	--

Description

This function checks whether the inputs have the correct format

Usage

```
check.functions(pnull, rnull, phat = function(x) -99, vals, x)
```

Arguments

pnull	cdf under the null hypothesis
rnull	routine to generate data under the null hypothesis
phat	=function(x) -99, function to estimate parameters from the data, or -99
vals	vector of discrete values
x	data

chi_power_cont	<i>This function finds the power of various chi-square tests for continuous data</i>
----------------	--

Description

This function finds the power of various chi-square tests for continuous data

Usage

```
chi_power_cont(
  pnull,
  ralt,
  param_alt,
  qnull = NA,
  phat = function(x) -99,
  w = function(x) -99,
  alpha = 0.05,
  Range = c(-99999, 99999),
  B = 1000,
  nbins = c(50, 10),
  rate = 0,
  minexpcount = 5,
  ChiUsePhat = TRUE
)
```

Arguments

pnull	function to find cdf under null hypothesis
ralt	function to generate data under alternative hypothesis
param_alt	vector of parameter values for distribution under alternative hypothesis
qnull	=NA function to find quantiles under null hypothesis, if available
phat	=function(x) -99, function to estimate parameters
w	=function(x) -99, optional weight function
alpha	=0.05, the level of the hypothesis test
Range	=c(-99999, 99999) limits of possible observations, if any
B	=1000 number of simulation runs to find power
nbins	=c(50,10), number of bins for chi square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
minexpcount	=5 minimal expected bin count required
ChiUsePhat	=TRUE, if TRUE param is estimated parameters and no minimization is used

Value

A numeric matrix of power values.

chi_power_disc	<i>This function finds the power of various chi-square tests for continuous data</i>
----------------	--

Description

This function finds the power of various chi-square tests for continuous data

Usage

```
chi_power_disc(
  pnull,
  ralt,
  param_alt,
  phat = function(x) -99,
  alpha = 0.05,
  B = 1000,
  nbins = c(50, 10),
  rate = 0,
  minexpcount = 5,
  ChiUsePhat = TRUE
)
```

Arguments

pnull	function to find cdf under null hypothesis
ralt	function to generate data under alternative hypothesis
param_alt	vector of parameter values for distribution under alternative hypothesis
phat	=function(x) -99, routine to estimate parameters
alpha	=0.05, the level of the hypothesis test
B	=1000 number of simulation runs to find power
nbins	=c(50,10), number of bins for chi square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
minexpcount	=5 minimal expected bin count required
ChiUsePhat	= TRUE, should chi square use minimum chi square method?

Value

A numeric matrix of power values.

chi_test_cont	<i>This function performs a number of chi-square gof tests for continuous data</i>
---------------	--

Description

This function performs a number of chi-square gof tests for continuous data

Usage

```
chi_test_cont(
  x,
  pnull,
  w = function(x) -99,
  phat = function(x) -99,
  qnull = NA,
  nbins = c(50, 10),
  rate = 0,
  Range = c(-99999, 99999),
  minexpcount = 5,
  ChiUsePhat = TRUE,
  allbins
)
```

Arguments

x	data set
pnull	cdf under the null hypothesis
w	function to find weights of observations, returns -99 if data is unweighted
phat	=function(x) -99, estimated parameters, or starting values of multi-D minimum chi square minimization, or -99 if no estimation is done
qnull	=NA quantile function, if available
nbins	=c(50, 10) number of bins for chi-square tests
rate	=0, rate of Poisson if sample size is random
Range	=c(-99999, 99999) limits of possible observations, if any
minexpcount	=5 minimal expected bin count required
ChiUsePhat	=TRUE, if TRUE param is estimated parameters and no minimization is used
allbins	set of bins to use

Value

A numeric matrix of test statistics, degrees of freedom and p.values

chi_test_disc	<i>This function performs a number of chi-square gof tests for continuous data</i>
---------------	--

Description

This function performs a number of chi-square gof tests for continuous data

Usage

```
chi_test_disc(
  x,
  pnull,
  phat = function(x) -99,
  nbins = c(50, 10),
  rate = 0,
  minexpcount = 5,
  ChiUsePhat = TRUE,
  allbins
)
```

Arguments

x	data set
pnull	cdf under the null hypothesis
phat	=function(x) -99, function to estimate parameters, or starting values of multi-D minimum chi square minimization, or -99 if no parameters are estimated
nbins	=c(50, 10) number of bins for chi-square tests
rate	=0, rate of Poisson if sample size is random
minexpcount	=5 minimal expected bin count required
ChiUsePhat	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
allbins	set of bins to use

Value

A numeric matrix of test statistics, degrees of freedom and p.values

gof_power	<i>Find the power of various gof tests for continuous data.</i>
-----------	---

Description

Find the power of various gof tests for continuous data.

Usage

```
gof_power(
  pnull,
  vals = NA,
  rnull,
  ralt,
  param_alt,
  w = function(x) -99,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  alpha = 0.05,
  Range = c(-Inf, Inf),
  B = 1000,
  nbins = c(50, 10),
  rate = 0,
  maxProcessor,
  minexpcount = 5,
  ChiUsePhat = TRUE
)
```

Arguments

<code>pnull</code>	function to find cdf under null hypothesis
<code>vals</code>	=NA, values of rv, if data is discrete, NA if data is continuous
<code>rnull</code>	function to generate data under null hypothesis
<code>ralt</code>	function to generate data under alternative hypothesis
<code>param_alt</code>	vector of parameter values for distribution under alternative hypothesis
<code>w</code>	(Optional) function to calculate weights, returns -99 if no weights
<code>phat</code>	=function(x) -99 function to estimate parameters from the data, or -99
<code>TS</code>	user supplied function to find test statistics
<code>TSextra</code>	=NA, list provided to TS
<code>alpha</code>	=0.05, the level of the hypothesis test
<code>Range</code>	=c(-Inf, Inf) limits of possible observations, if any
<code>B</code>	=1000 number of simulation runs
<code>nbins</code>	=c(100,10), number of bins for chi square tests.
<code>rate</code>	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
<code>maxProcessor</code>	maximum of number of processors to use, 1 if no parallel processing is needed or number of cores-1 if missing
<code>minexpcount</code>	=5 minimal expected bin count required
<code>ChiUsePhat</code>	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.

Value

A numeric matrix of power values.

Examples

```
# Power of tests when null hypothesis specifies the standard normal distribution but
# true data comes from a normal distribution with mean different from 0.
pnull = function(x) pnorm(x)
rnull = function() rnorm(50)
ralt = function(mu) rnorm(50, mu)
TSextra = list(qnull=function(x) qnorm(x))
gof_power(pnull, NA, rnull, ralt, c(0.25, 0.5), TSextra=TSextra, B=200)
# Power of tests when null hypothesis specifies normal distribution and
# mean and standard deviation are estimated from the data.
# Example is not run because it takes several minutes.
# true data comes from a normal distribution with mean different from 0.
pnull = function(x, p=c(0, 1)) pnorm(x, p[1], ifelse(p[2]>0.001, p[2], 0.001))
rnull = function(p=c(0, 1)) rnorm(50, p[1], ifelse(p[2]>0.001, p[2], 0.001))
phat = function(x) c(mean(x), sd(x))
TSextra = list(qnull = function(x, p=c(0, 1)) qnorm(x, p[1],
  ifelse(p[2]>0.001, p[2], 0.001)))
gof_power(pnull, NA, rnull, ralt, c(0, 1), phat=phat, TSextra=TSextra, B=200)
# Power of tests when null hypothesis specifies Poisson rv with rate 100 and
```



```

# true rate is 100.5
vals = 0:250
pnull = function() ppois(0:250, 100)
rnull =function () table(c(0:250, rpois(1000, 100)))-1
ralt =function (p) table(c(0:250, rpois(1000, p)))-1
gof_power(pnull, vals, rnull, ralt, param_alt=100.5, B=200)
# Power of tests when null hypothesis specifies a Binomial n=10 distribution
# with the success probability estimated
vals = 0:10
pnull=function(p) pbinom(0:10, 10, ifelse(0<p&p<1, p, 0.001))
rnull=function(p) table(c(0:10, rbinom(1000, 10, ifelse(0<p&p<1, p, 0.001))))-1
ralt=function(p) table(c(0:10, rbinom(1000, 10, p)))-1
phat=function(x) mean(rep(0:10,x))/10
gof_power(pnull, vals, rnull, ralt, c(0.5, 0.6), phat=phat, B=200)

```

gof_power_cont

*Find the power of various gof tests for continuous data.***Description**

Find the power of various gof tests for continuous data.

Usage

```

gof_power_cont(
  pnull,
  rnull,
  ralt,
  param_alt,
  w = function(x) -99,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  alpha = 0.05,
  Range = c(-Inf, Inf),
  B = 1000,
  nbins = c(100, 10),
  rate = 0,
  maxProcessor,
  minexpcount = 5,
  ChiUsePhat = TRUE
)

```

Arguments

pnull	function to find cdf under null hypothesis
rnull	function to generate data under null hypothesis

ralt	function to generate data under alternative hypothesis
param_alt	vector of parameter values for distribution under alternative hypothesis
w	=function(x) -99, function to calculate weights, returns -99 if no weights
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
TSextra	=NA, list provided to TS
alpha	=0.05, the level of the hypothesis test
Range	=c(-Inf, Inf) limits of possible observations, if any
B	=1000 number of simulation runs
nbins	=c(100,10), number of bins for chi square tests.
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
maxProcessor	maximum of number of processors to use, 1 if no parallel processing is needed or number of cores-1 if missing
minexpcount	=5 minimal expected bin count required
ChiUsePhat	=TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.

Value

A numeric matrix of power values.

gof_power_disc *Find the power of various gof tests for discrete data.*

Description

Find the power of various gof tests for discrete data.

Usage

```
gof_power_disc(
  pnull,
  rnull,
  vals,
  ralt,
  param_alt,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  alpha = 0.05,
  B = 1000,
  nbins = c(100, 10),
```

```

    rate = 0,
    maxProcessor,
    minexpcount = 5,
    ChiUsePhat = TRUE
  )

```

Arguments

<code>pnull</code>	cumulative distribution function under the null hypothesis
<code>rnull</code>	a function to generate data under null hypothesis
<code>vals</code>	values of discrete rv.
<code>ralt</code>	function to generate data under alternative hypothesis
<code>param_alt</code>	vector of parameter values for distribution under alternative hypothesis
<code>phat</code>	=function(x) -99, function to estimate parameters from the data, -99 if no parameters are estimated
<code>TS</code>	user supplied function to find test statistics, if any
<code>TSextra</code>	=NA, list passed to TS, if desired
<code>alpha</code>	=0.05, the level of the hypothesis test
<code>B</code>	=1000 number of simulation runs
<code>nbins</code>	=c(100, 10) number of bins for chi square tests
<code>rate</code>	rate of Poisson if sample size is random
<code>maxProcessor</code>	maximum of number of processors to use, 1 if no parallel processing is needed or number of cores-1 if missing
<code>minexpcount</code>	=5 minimal number of expected counts in each bin for chi square tests
<code>ChiUsePhat</code>	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.

Value

A numeric matrix of power values.

`gof_test`

This function performs a number of gof tests

Description

This function performs a number of gof tests

Usage

```

gof_test(
  x,
  vals = NA,
  pnull,
  rnull,
  w = function(x) -99,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  nbins = c(50, 10),
  rate = 0,
  Range = c(-Inf, Inf),
  B = 5000,
  minexpcount = 5,
  ChiUsePhat = TRUE,
  maxProcessor = 1,
  doMethods = "all"
)

```

Arguments

x	data set
vals	=NA, values of discrete RV, or NA if data is continuous
pnull	cdf under the null hypothesis
rnull	routine to generate data under the null hypothesis
w	(Optional) function to calculate weights, returns -99 if no weights
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
TSextra	=NA, list passed to TS, if desired, or NA
nbins	=c(100, 10) number of bins for chi-square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
Range	=c(-Inf, Inf) limits of possible observations, if any, for chi-square tests
B	=5000 number of simulation runs
minexpcount	=5 minimal expected bin count required
ChiUsePhat	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
maxProcessor	=1, number of processors to use in parallel processing.
doMethods	Methods to include in tests

Value

A list with vectors of test statistics and p.values

Examples

```

# Tests to see whether data comes from a standard normal distribution.
pnull = function(x) pnorm(x)
rnull = function() rnorm(100)
x = rnorm(100)
gof_test(x, NA, pnull, rnull, B=500)
# Tests to see whether data comes from a normal distribution with standard deviation 1
# and the mean estimated.
pnull=function(x, m) pnorm(x, m)
rnull=function(m) rnorm(100, m)
TSextra = list(qnull=function(x, m=0) qnorm(x, m),
              pnull=function(x, m=0) pnorm(x, m), phat=function(x) mean(x))
phat=function(x) mean(x)
x = rnorm(100, 1, 2)
gof_test(x, NA, pnull, rnull, phat=phat, TSextra=TSextra, B=500)
# Tests to see whether data comes from a binomial (10, 0.5) distribution.
vals=0:10
pnull = function() pbinom(0:10, 10, 0.5)
rnull = function() table(c(0:10, rbinom(1000, 10, 0.5)))-1
x = rnull()
gof_test(x, vals, pnull, rnull, doMethods="all", B=500)
# Tests to see whether data comes from a binomial distribution with
# the success probability estimated from the data.
pnull = function(p=0.5) pbinom(0:10, 10, ifelse(p>0&&p<1, p, 0.001))
rnull = function(p=0.5) table(c(0:10, rbinom(1000, 10,
                                     ifelse(p>0&&p<1, p, 0.001))))-1
phat=function(x) mean(rep(0:10,x))/10
gof_test(x, vals, pnull, rnull, phat=phat, B=500)

```

gof_test_adjusted_pvalue

This function performs a number of gof tests and finds the adjusted p value for the combined test

Description

This function performs a number of gof tests and finds the adjusted p value for the combined test

Usage

```

gof_test_adjusted_pvalue(
  x,
  vals = NA,
  pnull,
  rnull,
  w = function(x) -99,
  phat = function(x) -99,
  TS,

```

```

  TSextra = NA,
  nbins = c(50, 10),
  rate = 0,
  Range = c(-Inf, Inf),
  B = c(5000, 1000),
  minexpcount = 5,
  ChiUsePhat = TRUE,
  doMethods
)

```

Arguments

x	data set
vals	=NA, values of discrete RV, or NA if data is continuous
pnull	cdf under the null hypothesis
rnull	routine to generate data under the null hypothesis
w	(Optional) function to calculate weights, returns -99 if no weights
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
TSextra	=NA, list passed to TS, if desired, or NA
nbins	=c(100, 10) number of bins for chi-square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
Range	=c(-Inf, Inf) limits of possible observations, if any, for chi-square tests
B	=c(5000,1000) number of simulation runs for individual and for adjusted p values
minexpcount	=5 minimal expected bin count required
ChiUsePhat	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
doMethods	Methods to include in tests

Value

None

Examples

```

# Tests to see whether data comes from a standard normal distribution.
pnull = function(x) pnorm(x)
rnull = function() rnorm(100)
x = rnorm(100)
gof_test_adjusted_pvalue(x, NA, pnull, rnull, B=c(500, 200))
# Tests to see whether data comes from a normal distribution with standard deviation 1
# and the mean estimated.
pnull=function(x, m) pnorm(x, m)
rnull=function(m) rnorm(100, m)

```

```

TSextra = list(qnull=function(x, m=0) qnorm(x, m),
              pnull=function(x, m=0) pnorm(x, m), phat=function(x) mean(x))
phat=function(x) mean(x)
x = rnorm(100, 1, 2)
gof_test_adjusted_pvalue(x, NA, pnull, rnull, phat=phat, TSextra=TSextra, B=c(500, 200))
# Tests to see whether data comes from a binomial (10, 0.5) distribution.
vals=0:10
pnull = function() pbinom(0:10, 10, 0.5)
rnull = function() table(c(0:10, rbinom(1000, 10, 0.5)))-1
x = rnull()
gof_test_adjusted_pvalue(x, vals, pnull, rnull, B=c(500, 200))
# Tests to see whether data comes from a binomial distribution with
# the success probability estimated from the data.
pnull = function(p=0.5) pbinom(0:10, 10, ifelse(p>0&&p<1, p, 0.001))
rnull = function(p=0.5) table(c(0:10, rbinom(1000, 10,
                                     ifelse(p>0&&p<1, p, 0.001))))-1
phat=function(x) mean(rep(0:10,x))/10
gof_test_adjusted_pvalue(x, vals, pnull, rnull, phat=phat, B=c(500, 200))

```

gof_test_cont

This function performs a number of gof tests for continuous data

Description

This function performs a number of gof tests for continuous data

Usage

```

gof_test_cont(
  x,
  pnull,
  rnull,
  w = function(x) -99,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  nbins = c(50, 10),
  rate = 0,
  Range = c(-Inf, Inf),
  B = 5000,
  minexpcount = 5,
  ChiUsePhat = TRUE,
  maxProcessor = 1,
  doMethods = "all"
)

```

Arguments

x	data set
pnull	cdf under the null hypothesis
rnull	routine to generate data under the null hypothesis
w	(Optional) function to calculate weights, returns -99 if no weights
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
TSextra	=NA, list passed to TS, if desired
nbins	=c(50, 10) number of bins for chi-square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
Range	=c(-Inf, Inf) limits of possible observations, if any, for chi-square tests
B	=5000 number of simulation runs
minexpcount	=5 minimal expected bin count required
ChiUsePhat	=TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
maxProcessor	=1, number of processors to use in parallel processing. If missing single processor is used.
doMethods	Methods to include in tests

Value

A list with vectors of test statistics and p.values

gof_test_cont_adj	<i>This function performs a number of gof tests for continuous data and finds the adjusted p value</i>
-------------------	--

Description

This function performs a number of gof tests for continuous data and finds the adjusted p value

Usage

```
gof_test_cont_adj(
  x,
  pnull,
  rnull,
  w = function(x) -99,
  phat = function(x) 0,
  TS,
  TSextra = NA,
```



```

nbins = c(50, 10),
rate = 0,
Range = c(-Inf, Inf),
B = c(5000, 1000),
minexpcount = 5,
ChiUsePhat = TRUE,
doMethods = c("W", "ZC", "AD", "ES-s-P")
)

```

Arguments

x	data set
pnull	cdf under the null hypothesis
rnull	routine to generate data under the null hypothesis
w	=function(x) =99 (Optional) function to calculate weights, returns -99 if no weights
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
TSextra	=NA, list passed to TS, if desired
nbins	=c(50, 10) number of bins for chi-square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
Range	=c(-Inf, Inf) limits of possible observations, if any, for chi-square tests
B	=c(5000,1000) number of simulation runs for p values and for p value distribution
minexpcount	=5 minimal expected bin count required
ChiUsePhat	=TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
doMethods	Methods to include in tests

Value

None

gof_test_disc

This function performs a number of gof tests for discrete data.

Description

This function performs a number of gof tests for discrete data.

Usage

```

gof_test_disc(
  x,
  pnull,
  rnull,
  vals,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  nbins = c(50, 10),
  rate = 0,
  B = 5000,
  minexpcount = 5,
  ChiUsePhat = TRUE,
  maxProcessor = 1,
  doMethods = "Default"
)

```

Arguments

x	data set (the counts)
pnull	cumulative distribution function under the null hypothesis
rnull	routine to generate data under the null hypothesis
vals	a vector of values of discrete random variables
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	=NA, user supplied function to find test statistics
TSextra	=NA, list passed to TS, if desired
nbins	=c(50, 10) number of bins for chi-square tests
rate	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
B	=5000 number of simulation runs
minexpcount	=5 minimal expected bin count required
ChiUsePhat	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
maxProcessor	=1, number of processors to use in parallel processing. If missing single processor is used.
doMethods	Methods to include in tests

Value

A numeric matrix of test statistics and p.values

`gof_test_disc_adj` *This function performs a number of gof tests for discrete data and finds the adjusted p value*

Description

This function performs a number of gof tests for discrete data and finds the adjusted p value

Usage

```
gof_test_disc_adj(
  x,
  pnull,
  rnull,
  vals,
  phat = function(x) -99,
  TS,
  TSextra = NA,
  nbins = c(50, 10),
  rate = 0,
  B = c(5000, 1000),
  minexpcount = 5,
  ChiUsePhat = TRUE,
  doMethods = c("Wassp1", "W", "AD", "s-P")
)
```

Arguments

<code>x</code>	data set (the counts)
<code>pnull</code>	cumulative distribution function under the null hypothesis
<code>rnull</code>	routine to generate data under the null hypothesis
<code>vals</code>	a vector of values of discrete random variables
<code>phat</code>	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
<code>TS</code>	=NA, user supplied function to find test statistics
<code>TSextra</code>	=NA, list passed to TS, if desired
<code>nbins</code>	=c(50, 10) number of bins for chi-square tests
<code>rate</code>	=0 rate of Poisson if sample size is random, 0 if sample size is fixed
<code>B</code>	=c(5000, 1000) number of simulation runs for p values and for adjusted p value
<code>minexpcount</code>	=5 minimal expected bin count required
<code>ChiUsePhat</code>	= TRUE, if TRUE param is estimated parameter, otherwise minimum chi square method is used.
<code>doMethods</code>	Methods to include in tests

Value

A numeric matrix of test statistics and p.values

make_bins_cont	<i>This function creates several type of bins for continuous data</i>
----------------	---

Description

This function creates several type of bins for continuous data

Usage

```
make_bins_cont(  
  x,  
  pnull,  
  qnull = NA,  
  phat = function(x) -99,  
  DataBased = FALSE,  
  nbins = c(50, 10),  
  minexpcount = 5,  
  Range = c(-99999, 99999)  
)
```

Arguments

x	data set
pnull	cdf under the null hypothesis
qnull	=NA quantile function, if available
phat	=function(x) -99 parameters for pnull
DataBased	=FALSE bins based on data, not expected counts
nbins	=c(50, 10) number of bins
minexpcount	=5 smallest expected count per bin
Range	=c(-99999, 99999) limits of possible observations, if any

Value

A list of bins and bin probabilities

make_bins_disc	<i>This function creates several types of bins for discrete data</i>
----------------	--

Description

This function creates several types of bins for discrete data

Usage

```
make_bins_disc(
  x,
  pnull,
  phat = function(x) -99,
  nbins = c(50, 10),
  minexpcount = 5
)
```

Arguments

x	counts
pnull	cumulative distribution function
phat	=function(x) -99, function to estimated parameters, or -99
nbins	=c(50, 10) number of bins
minexpcount	=5 smallest expected count per bin

Value

A list of indices

newTSdisc	<i>a local function needed for the vignette</i>
-----------	---

Description

a local function needed for the vignette

Usage

```
newTSdisc(x, pnull, param, vals)
```

Arguments

x	An integer vector.
pnull	cdf.
param	parameters for pnull in case of parameter estimation.
vals	A numeric vector with the values of the discrete rv.

Value

A vector with test statistics

plot_power	<i>This function draws the power graph, with curves sorted by the mean power and smoothed for easier reading.</i>
------------	---

Description

This function draws the power graph, with curves sorted by the mean power and smoothed for easier reading.

Usage

```
plot_power(pwr, xname = " ", title, Smooth = TRUE, span = 0.25)
```

Arguments

pwr	a matrix of power values, usually from the twosample_power command
xname	Name of variable on x axis
title	(Optional) title of graph
Smooth	=TRUE lines are smoothed for easier reading
span	=0.25bandwidth of smoothing method

Value

plt, an object of class ggplot.

power_cont_R	<i>Find the power of various gof tests for continuous data.</i>
--------------	---

Description

Find the power of various gof tests for continuous data.

Usage

```
power_cont_R(
  pnull,
  rnull,
  ralt,
  param_alt,
  phat = function(x) -99,
  TS,
  typeTS,
  TSextra,
  alpha = 0.05,
  B = 1000,
  maxProcessor
)
```

Arguments

pnull	function to find cdf under null hypothesis
rnull	function to generate data under null hypothesis
ralt	function to generate data under alternative hypothesis
param_alt	vector of parameter values for distribution under alternative hypothesis
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
typeTS	format of TS routine
TSextra	list provided to TS
alpha	=0.05, the level of the hypothesis test
B	=1000 number of simulation runs
maxProcessor	maximum of number of processors to use, 1 if no parallel processing is needed or number of cores-1 if missing

Value

A numeric matrix of power values

power_disc_R

Find the power of various gof tests for discrete data.

Description

Find the power of various gof tests for discrete data.

Usage

```
power_disc_R(
  pnull,
  rnull,
  vals,
  ralt,
  param_alt,
  phat = function(x) -99,
  TS,
  typeTS,
  TSextra,
  alpha = 0.05,
  B = 1000,
  maxProcessor
)
```

Arguments

pnull	function to find cdf under null hypothesis
rnull	function to generate data under null hypothesis
vals	values of discrete distribution
ralt	function to generate data under alternative hypothesis
param_alt	vector of parameter values for distribution under alternative hypothesis
phat	=function(x) -99, function to estimate parameters from the data, or -99 if no parameters are estimated
TS	user supplied function to find test statistics, if any
typeTS	format of TS routine
TSextra	list provided to TS
alpha	=0.05, the level of the hypothesis test
B	=1000 number of simulation runs
maxProcessor	maximum of number of processors to use, 1 if no parallel processing is needed or number of cores-1 if missing

Value

A numeric matrix of power values

power_newtest	<i>This function estimates the power of test routines that calculate p value(s)</i>
---------------	---

Description

This function estimates the power of test routines that calculate p value(s)

Usage

```
power_newtest(
  TS,
  vals = NA,
  pnull,
  ralt,
  param_alt,
  phat,
  TSextra,
  alpha = 0.05,
  B = 1000
)
```

Arguments

TS	routine to calculate test statistics.
vals	=NA if data is discrete, a vector of possible values
pnull	routine to calculate the cdf under the null hypothesis
ralt	generate data under alternative hypothesis
param_alt	values of parameter under the alternative hypothesis.
phat	function to estimate parameters, function(x) -99 if no parameter estimation
TSextra	list (possibly) passed to TS
alpha	=0.05 type I error.
B	= 1000 number of simulation runs to estimate the power.

Value

A matrix of power values

power_studies_results *power_studies_results*

Description

the results of the included power studies

Usage

power_studies_results

Format

'power_studies_results':

A list of matrices with powers

pvaluecdf *pvaluecdf*

Description

the info needed to draw a graph

Usage

pvaluecdf

Format

'pvaluecdf':

A matrix

run.studies

This function runs the case studies included in the package

Description

This function runs the case studies included in the package

Usage

```
run.studies(
  TS,
  study,
  TSextra = list(aaa = 1),
  With.p.value = FALSE,
  BasicComparison = TRUE,
  nsample = 500,
  alpha = 0.05,
  param_alt,
  maxProcessor,
  B = 1000
)
```

Arguments

TS	routine to calculate test statistic(s) or p value(s).
study	either the name of the study, or its number. If missing all the studies are run.
TSextra	=list(aaa=1), list passed to TS.
With.p.value	=FALSE does user supplied routine return p values?
BasicComparison	=TRUE if true compares tests on one default value of parameter of the alternative distribution.
nsample	= 500, desired sample size.
alpha	=0.05 type I error
param_alt	(list of) values of parameter under the alternative hypothesis. If missing included values are used.
maxProcessor	number of cores to use for parallel programming
B	= 1000 number of simulation runs

Value

A (list of) matrices of p.values

Examples

```
# New test is a simple chi-square test:
chitest=function(x, pnull, param, TSextra) {
  nbins=TSextra$nbins
  bins=quantile(x, (0:nbins)/nbins)
  O=hist(x, bins, plot=FALSE)$counts
  if(param[1]!=-99) { #with parameter estimation
    E=length(x)*diff(pnull(bins, param))
    chi=sum((O-E)^2/E)
    pval=1-pchisq(chi, nbins-1-length(param))
  }
  else {
    E=length(x)*diff(pnull(bins))
    chi=sum((O-E)^2/E)
    pval=1-pchisq(chi, nbins-1)
  }
  out=ifelse(TSextra$statistic, chi, pval)
  names(out)="ChiSquare"
  out
}
TSextra=list(nbins=10, statistic=FALSE) # Use 10 bins, test routine returns p-value
run.studies(chitest, TSextra=TSextra, With.p.value=TRUE, maxProcessor=1, B=200)
```

signif.digits

This function does some rounding to nice numbers

Description

This function does some rounding to nice numbers

Usage

```
## S3 method for class 'digits'
signif(x, d = 4)
```

Arguments

x	a list of two vectors
d	=4 number of digits to round to

Value

A list with rounded vectors

timecheck	<i>estimate run time function</i>
-----------	-----------------------------------

Description

estimate run time function

Usage

```
timecheck(x, pnull, phatx, wx, TS, typeTS, TSextra)
```

Arguments

x	data set
pnull	function to find cdf under null hypothesis
phatx	parameter estimates
wx	vector of wights
TS	test statistic
typeTS	format of TS
TSextra	additional info TS

Value

Mean computation time

TS_cont	<i>Find test statistics for continuous data</i>
---------	---

Description

Find test statistics for continuous data

Usage

```
TS_cont(x, pnull, param, qnull)
```

Arguments

x	A numeric vector.
pnull	cdf.
param	parameters for pnull in case of parameter estimation.
qnull	An R function, the quantile function under the null hypothesis.

Value

A numeric vector with test statistics

`TS_disc`*Find test statistics for discrete data*

Description

Find test statistics for discrete data

Usage

```
TS_disc(x, pnull, param, vals)
```

Arguments

<code>x</code>	An integer vector.
<code>pnull</code>	cdf.
<code>param</code>	parameters for <code>pnull</code> in case of parameter estimation.
<code>vals</code>	A numeric vector with the values of the discrete rv.

Value

A vector with test statistics

Index

* datasets

- power_studies_results, 26
- pvaluecdf, 26

- case.studies, 2
- check.functions, 3
- chi_power_cont, 3
- chi_power_disc, 4
- chi_test_cont, 5
- chi_test_disc, 6

- gof_power, 7
- gof_power_cont, 9
- gof_power_disc, 10
- gof_test, 11
- gof_test_adjusted_pvalue, 13
- gof_test_cont, 15
- gof_test_cont_adj, 16
- gof_test_disc, 17
- gof_test_disc_adj, 19

- make_bins_cont, 20
- make_bins_disc, 21

- newTSdisc, 21

- plot_power, 22
- power_cont_R, 22
- power_disc_R, 23
- power_newtest, 25
- power_studies_results, 26
- pvaluecdf, 26

- run.studies, 27

- signif.digits, 28

- timecheck, 29
- TS_cont, 29
- TS_disc, 30