

# Package: simgof (via r-universe)

October 24, 2024

**Title** Simultaneous Goodness-of-Fits Tests

**Version** 1.0.2

**Description** Routine that allows the user to run several goodness-of-fit tests. It also combines the tests and returns a properly adjusted family-wise p value. Details can be found in [arXiv:2007.04727](https://arxiv.org/abs/2007.04727).

**Depends** R (>= 3.1.0)

**Imports** ddst, stats, graphics

**License** GPL-2

**Encoding** UTF-8

**NeedsCompilation** no

**LazyData** true

**RoxygenNote** 7.1.1

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

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

**Date/Publication** 2021-01-27 09:00:02 UTC

**Repository** <https://wolfgangrolke.r-universe.dev>

**RemoteUrl** <https://github.com/cran/simgof>

**RemoteRef** HEAD

**RemoteSha** 00285288581932f0b96179ad9183bf2eaaad13a34

## Contents

chisquare.test . . . . .	2
simgof.test . . . . .	2
spreadout . . . . .	3
TS . . . . .	4

<b>Index</b>	<b>6</b>
--------------	----------

---

chisquare.test	<i>chisquare.test</i>
----------------	-----------------------

---

**Description**

This function does the chisquare test

**Usage**

```
chisquare.test(x, case, which = "RGd")
```

**Arguments**

x	data set
case	setup info
which	type of binning (either RGd, Equal Size or Equal Prob)

**Value**

A numeric vector of length 1 with the value of the chi-square statistic.

**Examples**

```
case <- list(B=1000, param = NULL, n = 1000, pnull = function(x, param) punif(x),
            rnull = function(n, param) runif(n), qnull = function(x, param) qunif(x),
            est.mle = function(x) NA, nbins = 10)
x <- runif(1000)
chisquare.test(x, case)
```

---

simgof.test	<i>simgof.test</i>
-------------	--------------------

---

**Description**

This function performs a number of gof tests and rejects the null if any of the tests does so. Then it finds the adjusted p-value.

**Usage**

```
simgof.test(
  x,
  pnull,
  rnull,
  qnull = function(x) NULL,
  do.estimation = TRUE,
  estimate = function(x) NULL,
```

```

include.methods = c(rep(TRUE, 7), rep(FALSE, 9)),
B = 10000,
lambda,
nbins = NULL
)

```

### Arguments

x	data set
pnull	distribution function under the null hypothesis
rnull	routine to generate data under the null hypothesis
qnull	quantile function under the null hypothesis
do.estimation	TRUE if parameters are to be estimated
estimate	routine for parameter estimation
include.methods	which methods should be used, a vector of length 16 of T/F
B	=10000 number of simulation runs
lambda	rate of Poisson if sample size is random
nbins	number of bins for chisquare test

### Value

A numeric vector of p values

### Examples

```

x <- runif(1000)
pnull <- function(x) x
rnull <- function(n) runif(n)
qnull <- function(x) x
simgof.test(x, pnull, rnull, qnull, FALSE, B=500)
x <- rnorm(1000, 100, 20)
pnull <- function(x, param) pnorm(x, param[1], param[2])
rnull <- function(n, param) rnorm(x, param[1], param[2])
qnull <- function(x, param) qnorm(x, param[1], param[2])
estimate <- function(x) c(mean(x), sd(x))
simgof.test(x, pnull, rnull, qnull, TRUE, estimate, B=500)

```

---

spreadout

*spreadout*

---

### Description

This function unbins data. If qnull is given it uses quantiles, otherwise uniform

**Usage**

```
spreadout(x, case)
```

**Arguments**

x	data set
case	setup info

**Value**

A numeric vector of observations without ties.

**Examples**

```
case <- list(B=1000, param = NULL, n = 1000, pnull = function(x, param) punif(x),
  rnull = function(n, param) runif(n), qnull = function(x, param) qunif(x),
  est.mle = function(x) NA, nbins = 10)
y=runif(1000)
bins=seq(0, 1, length=11)
counts=hist(y, bins, plot=FALSE)$counts
x=list(bins=bins,counts=counts)
spreadout(x, case)
```

---

 TS

---

 TS
 

---

**Description**

This function finds various gof statistics

**Usage**

```
TS(x, case)
```

**Arguments**

x	data
case	setup info

**Value**

A numeric vector with the values of various test statistics.

**Examples**

```
case <- list(B=1000, param = NULL, n = 1000, pnull = function(x, param)
  punif(x), rnull = function(n, param) runif(n), qnull = function(x, param)
  qunif(x), est.mle = function(x) NA, nbins = 10)
case$methods=c("KS", "AD", "CdM", "W", "ZA", "ZK", "ZC")
x <- runif(1000)
TS(x, case)
```

# Index

chisquare.test, 2

sigof.test, 2

spreadout, 3

TS, 4