Package 'HWxtest'

May 31, 2019

Type Package

Title Exact Tests for Hardy-Weinberg Proportions

Version 1.1.9

Date 2019-05-29

Author Bill Engels <wrengels@wisc.edu>

Maintainer Bill Engels <wrengels@wisc.edu>

Description Tests whether a set of genotype counts fits the HW expectations. Exact tests performed by an efficient algorithm. Included test statistics are likelihood ratio, probability, U-score and Pearson's X2.

VignetteBuilder knitr

Depends R (>= 2.14)

Suggests ggplot2, rmarkdown, knitr, RefManageR, adegenet, genetics, parallel

NeedsCompilation yes

License GPL

RoxygenNote 5.0.0

Repository CRAN

Date/Publication 2019-05-31 08:50:12 UTC

R topics documented:

`acount` *Find Approximate Number of Genotype Tables*

Description

Use `acount` to obtain the approximate number of genotype tables for a given set of allele counts. This method uses a normal approximation and is much faster than enumerating the tables with `xcount` but not as accurate.

Usage

```
acount(m)
```

Arguments

`m`	vector containing the numbers of alleles of each type. Length must be at least 2. All items are positive integers. It can also be a matrix of genotype counts, an object of type `genotype`, but not a vector of genotype counts.

Value

The approximate number of tables.

References

The methods are described by Engels, 2009. **Genetics** 183:1431.

See Also

`hwx.test`, `xcount`

Examples

```
# Allele counts from human Rh locus. Guo and Thompson, 1992, Figure 1
#
alleles <- c(15, 14, 11, 12, 2, 2, 1, 3)
acount(alleles)
# This approximation may be compared with the exact value of 250552020
#
ld <- c(6329, 319, 47, 2773, 75, 6702, 14, 2, 333)
acount(ld)
```

```
#
# This is an example where the number of tables is too large for a full enumeration.
```

`defaultHistobounds` *Functions to plot a histogram of test statistic*

Description

Running the function `hwx.test` can create data for a frequency distribution plot of one of the four test statistics provided the parameter `histobins` is positive. These plotting data are contained within the "hwtest" object generated by `hwx.test`. When that object is printed, a plot is drawn by `ggplot2`. If the user wishes to capture the `ggplot2` results, such as to use in making a composite figure, the "gg" object can be obtained by calling `makeHistogram`.

Usage

```
defaultHistobounds(ostats, statID, m)

makeHistogram(x, curveColor = "blue", color1 = "gray40",
  color2 = "lightcoral")
```

Arguments

`ostats`	Observed statistics for the 4 test measures, `LLR`, `Prob`, `U` and `Chisq`.
`statID`	Value 1-4 indicating which statistic to use for the plot.
`m`	vector of allele counts
`x`	output from `hwx.test`
`curveColor`	color for the asymptotic distribution curve
`color1`	The color for outcomes fitting the null distribution better than the observed
`color2`	The color for outcomes deviating from the null at leasst as much as observed. Area of `color2` is the P value.

Value

`defaultHistobounds` returns a vector containing the left and right boundaries for the x axis. This function is not normally called by the user

`makeHistobram` A graphic object of class "gg" or "ggplot" from `ggplot2`.

See Also

`hwx.test`

`fillUpper` *mungeData*

Description

Utility functions for handling genotype counts and arranging data

remove missing alleles

converts matrix to vector

Clears upper-right of matrix

Usage

```
fillUpper(gmat)

alleleCounts(gmat)

vec.to.matrix(gvec, alleleNames = "")

remove.missing.alleles(gmat)

matrix.to.vec(gmat)

clearUpper(gmat)

df.to.matrices(df, sep = "/")
```

Arguments

`gmat`	a matrix of non-negative integers representing genotype counts. In a matrix of genotype counts, `a[i,j]` and `a[j,i]` both represent the same heterozygote. Only the lower-left half of `gmat` is used. Numbers along the diagonal represent counts of the homozygotes.
`gvec`	vector containing `k(k+1)/2` genotype counts. All non-negative integers. Genotype counts should be in the order: `a11, a21, a22, a31, a32, ..., akk`
`alleleNames`	an optional list of names for the alleles. The length should be k
`df`	a dataframe containing individual genotypes. Each row represents an individual. The first column, named "pop" names the population. Each other column is named for a particular locus. The genotypes are as "123/124"
`sep`	For a dataframe, this is the separator character. typically "/"

Details

Interconvert between different formats for genotype counts.

Let `k` be the number of alleles:

- `clearUpper` fills the upper-right half of the kxk matrix with `NA`
- `fillUpper` makes the kxk matrix symmetrical by filling the upper-right half with numbers from the lower half.
- `vec.to.matrix` converts genotype counts in vector form and returns a matrix. The vector must have $k(k+1)/2$ non-negative integers.
- `matrix.to.vec` converts a kxk matrix of genotype counts to a vector of length $k(k+1)/2$
- `alleleCounts` returns a vector of length k containing the numbers of each allele. The sum of this vector will be twice the number of diploids in the sample.
- `remove.missing.alleles` returns a matrix with no `0`'s for allele counts
- `df.to.matrices` converts a data frame to a list of genotype count matrices. The data frame should be of the kind produced in the package `adegenet` with `genind2df`

none

none

Examples

```
gvec <- c(0,3,1,5,18,1,3,7,5,2)
gmat <- vec.to.matrix(gvec, alleleNames=letters[1:4])
alleleCounts(gmat)
```

`genepop.to.genind` *Imports a* `.txt` *file in* `GenePop` *format into an object of type* `genind`

Description

The main work is done by the function `adegenet::read.genepop`. However, that function requires text files with an extension of `.gen`, whereas such files usually have extension `.txt`. The sole purpose of this function is to work around the ".gen" requirement.

Usage

```
genepop.to.genind(name, quiet = TRUE, ncode = 3)
```

Arguments

`name`	the name of a file in `GenePop` format
`quiet`	whether a conversion message should be printed
`ncode`	Set to the number of characters per allele name

Value

an object of class `genind`

HWcases *This is a data file with some HW examples in matrix form*

Description

This is a data file with some HW examples in matrix form

References

Louis and Dempster, 1987

Guo and Thompson, 1992

Rousset 2007 – from the documentation of GenePop

hwdf *Construct a data frame from* hwx.test *output*

Description

If the hwx.test output has multiple populations and/or multiple loci, use this function to make a data frame to display the results in tabular form.

Usage

```
hwdf(hwlist, statName = NA, showN = TRUE, showk = TRUE,
  showMethod = TRUE, showSE = TRUE, showTables = TRUE,
  showTrials = TRUE, showStat = TRUE, showAsymptoticX2 = FALSE,
  showAsymptoticG2 = FALSE)
```

Arguments

hwlist	The output from a call to hwx.test
statName	gives you the option of changing which statistic's P value is reported
showN	whether to show a column of sample size (number of diploids in the sample)
showk	whether to show the number of alleles
showMethod	whether to show whether the exact or Monte Carlo method was used
showSE	whether to include the standard error for those tests which used the Monte Carlo method
showTables	whether to show the total number of tables examined when full enumeration (exact) method is used
showTrials	whether to show the number of random trials when Monte Carlo method is used
showStat	whether to show the observed statistic

showAsymptoticX2
: whether to include the asymptotic P value corresponding to the Pearson X^2 statistic

showAsymptoticG2
: whether to include the asymptotic P value for the LLR statistic

---

hwx.test *Test for HW by either full enumeration or Monte Carlo.*

Description

The hwx.test() function is the main function of the HWxtest package. This function produces a valid test for Hardy-Weinberg frequencies for virtually any set of genotype counts. It will use either a full-enumeration method in which all possible tables with the same allele numbers are examined, or a Monte Carlo test where a large number of random tables is examined. To decide which to use, it calls xcountCutoff to determine whether the number of tables to examine is greater than cutoff. If it is, then mtest is used. Otherwise xtest is used. The result is a robust test which will always provide a meaningful and accurate P value. Each table examined is compared with the observed counts according to each of four measures of fit: "LLR", "Prob", "U", or "Chisq" corresponding to the log-likelihood ratio, the null-hypothesis probability, the U-score or the Pearson X^2 value. It can also plot a histogram showing the distribution of any of these statistics.

Usage

```
hwx.test(c, method = "auto", cutoff = 1e+07, B = 1e+05,
  statName = "LLR", histobins = 0, histobounds = c(0, 0), showCurve = T,
  safeSecs = 100, detail = 2)
```

Arguments

c
: The genotype counts. You must provide the number of each genotype. So if there are k alleles, you need to include the number of each of the $k(k+1)/2$ genotypes. The format of x is somewhat flexible: It can be a square matrix, but only the lower-left half is used. It can be a vector of the observations in the order $a_1 1, a_2 1, a_2 2, a_3 1, ..., a_k k$. For compatability with the packages genetics and adegenet, it can also be an object of class genind, genotype, or a data.frame. If c contains multiple samples, the parallel package will be used in an attempt to employ multi-cores.

method
: Can be "auto", "exact" or "monte" to indicate the method to use. If "auto", the hwx.test will first check to see whether the total number of tables exceeds a cutoff specified by the parameter cutoff.

cutoff
: If method is set to "auto", then cutoff is used to decide whether to perform the test via the full enumeration or Monte Carlo method. If the number of tables is less than cutoff, then a full enumeration is performed. Otherwise the method will be Monte Carlo with B random trials.

B
: The number of trials to perform if Monte Carlo method is used

statName	can be "LLR", "Prob", "U", or "Chisq" depending on which one is to be ploted. Note that P values for all four are computed regardless of which one is specified with this parameter.
histobins	If 0, no histogram is plotted. If 1 or TRUE a histogram with 500 bins is plotted. If histobins is set to a number greater than 1, a histogram with histobins bins is plotted.
histobounds	A vector containing the left and right boundaries for the histogram's x axis. If you leave this as the default, c(0,0), then hwx.test will compute reasonable bounds to include most of the distribution.
showCurve	whether to show a blue curve indicating the asymptotic (chi squared) distribution. This only works for LLR and Chisq
safeSecs	After this many seconds the calculation will be aborted. This is a safety valve to prevent attempts to compute impossibly large sets of tables.
detail	Determines how much detail is printed. If it is set to 0, nothing is printed (useful if you use hwx.test programmatically.)

Value

Returns a list of class hwtest which includes the following items:

$ Pvalues	The four computed P values corresponding to the test statistics: LLR, Prob, U and Chisq in that order.
$ observed	The four observed statistics in the same order as above
$ ntrials	The number of tables examined during the calculation if done by Monte Carlo
$ tableCount	The total number of tables if done by full enumeration
$ genotypes	The input matrix of genotype counts
$ alleles	The allele counts m corresponding to the input genotype counts
$ statName	Which statistic to use for the histogram and in the p.value item
$ method	Which method was used, "exact" or "monte"
$ detail	An integer indicating how much detail to print. Use 0 for no printing
$ SE	vector with the standard error for each stat. Only applicable with Monte Carlo tests

References

The methods are described by Engels, 2009. **Genetics** 183:1431.

Examples

```
# Data from Louis and Dempster 1987 Table 2 and Guo and Thompson 1992 Figure 2:
c <- c(0,3,1,5,18,1,3,7,5,2)
hwx.test(c)
# To see a histogram of the LLR statistic:
hwx.test(c, histobins=TRUE)
# For a histogram of the U statistic and other details of the result:
hwx.test(c, statName="U", histobins=TRUE, detail=3)
```

listify — *Convert results of* hwx.test *to a single list of* hwtest *objects.*

Description

There are two main uses of listify. You can simplify a complex result from hwx.test containing multiple populations and multiple loci into a simple list of hwtest objects. At the same time, you have a chance to change the parameters detail and statName. Useful to get output from a test.

Usage

```
listify(hwlist, detail = NA, statName = NA)
```

Arguments

hwlist	the results of a call to hwx.test. It can be an hwtest object, a list of them or a list of lists of them.
detail	Used only if you wish to reset the detail of each object.
statName	Used only if you want to rest the statName of each object

Value

a list of hwtest objects, possibly with their detail and statName parameters reset

Examples

```
data(HWcases)
outcome <- hwx.test(HWcases, detail=4, statName="LLR")
listify(outcome, detail=1, statName="U")
```

mtest — *Performs an "exact" test using Monte Carlo trials for Hardy-Weinberg proportions*

Description

Given a set of genotype counts, mtest examines a large number of possible outcomes with the same set of allele counts. For each table, it computes four test statistics and compares them with the observed values. It returns the total probability of all tables with test statistics as "extreme" or more so than the observed. It can also plot a histogram of one of the statitistics if histobins is greater than zero. More about these four test statistics and other information can be found in the vignette. This function will not usually be called directly by the user. Instead, call hwx.test with method set to either "auto" or "monte".

Usage

```
mtest(c, ntrials = 1e+05, statName = "LLR", histobins = 0,
  histobounds = c(0, 0), showCurve = T, safeSecs = 100, detail = 2)
```

Arguments

c	A matrix containing the genotype counts. It should be a square matrix, but only the lower-left half is used.
ntrials	the number of random trials to perform
statName	can be "LLR", "Prob", "U", or "Chisq" depending on which one is to be ploted. Note that P values for all four are computed regardless of which one is specified with this parameter.
histobins	If 0 no histogram is plotted. If 1 or TRUE a histogram with 500 bins is plotted. If set to a number greater than 1, a histogram with histobins is plotted.
histobounds	A vector containing the left and right boundaries for the histogram's x axis. If you leave this as the default, c(0,0), then mtest will compute reasonable bounds to include most of the distribution.
showCurve	whether to show a blue curve indicating the asymptotic (chi squared) distribution. This only works for LLR and Chisq
safeSecs	After this many seconds the calculation will be aborted. This is a safety valve to prevent attempts to compute impossibly large sets of tables.
detail	Determines how much detail is printed. If it is set to 0, nothing is printed (useful if you use mtest programmatically.).

Value

mtest returns a list components

$ Pvalues	The four computed P values corresponding to the test statistics: LLR, Prob, U and Chisq in that order.
$ tableCount	placeholder
$ SE	Standard errors for the P values. These come from the binomial.
$ observed	The four observed statistics in the same order as above
$ ntrials	The number of tables examined during the calculation
$ genotypes	The input matrix of genotype counts
$ alleles	The allele counts m corresponding to the input genotype counts
$ statID	Which test statistic was used if a histogram was plotted
$ histobins	If greater than zero, the number of bins to use for the histogram
$ histobounds	The lower and upper limits of the test statistic in the histogram
$ histoData	Vector of $histobins$ values for the histogram
$ showCurve	Whether the asymptotic curve should be plotted with the histogram

References

The methods are described by Engels, 2009. **Genetics** 183:1431.

See Also

hwx.test

---

observedProb *Compute observed statistics for a genotype count matrix*

Description

#' Four measures of fit to Hardy-Weinberg for a given set of genotype counts may be computed.

- observedProb The probability of the observed set under the HW null and with the allele counts fixed.
- observedLLR The log-likelihood ratio of the observed set
- observedU The observed U-score. Positive values indicate an excess of homozygotes and negative ones imply too many heterozygotes
- observedX2 The classical "chi-squared" statistic

Usage

```
observedProb(c)

observedLLR(c)

observedU(c)

observedX2(c, returnExpected = F)
```

Arguments

c	Matrix of observed genotype counts. Each number should be a non-negative integer, and matrix is kxk.
returnExpected	Used in observedX2 to indicate whether a matrix of expected numbers should be returned instead.

Value

the observed statistic

Examples

```
t <- vec.to.matrix(c(0,3,1,5,18,1,3,7,5,2))
observedStats <- c(observedProb(t), observedLLR(t), observedU(t), observedX2(t))
```

p.value *Extract just the P value(s) from a Hardy-Weinberg test.*

Description

Use the p.value function to return just the P value(s) from the results of a call to hwx.test. If applied to a list of results, it will return a vector or matrix of P values. You can specify the statName as "LLR", "Prob", "U" or "Chisq". You can also apply p.value to a matrix or vector and it will attempt to use hwx.test to return a P value. However, it's usually preferable to use hwx.test directly.

Usage

```
p.value(x, statName = NA)
```

Arguments

x	The result of a call to hwx.test or a list of such results. It can also be the genotype counts in any of the same formats as accepted by hwx.test
statName	can be "LLR", "Prob", "U", or "Chisq"

Value

The P value

References

The methods are described by Engels, 2009. **Genetics** 183:1431.

Examples

```
data(HWcases)
testResults <- hwx.test(HWcases)
p.value(testResults)
p.value(testResults, statName="U")
```

print.hwtest *S3 Method for printing* hwtest *objects*

Description

Prints test results (hwtest) objects depending on how much detail is provided. If histogram data are present, ggplot2 is used to draw the plot by calling makeHistogram

Usage

```
## S3 method for class 'hwtest'
print(x, detail = NA, statName = NA, plotHisto = TRUE,
  ...)
```

Arguments

x	the results from a call to hwx.test
detail	0 for no print; 1 for P value only; 2 for all four P values; 3 to add data; 4 to add expected values
statName	which statistic to use
plotHisto	Indicate whether or not to plot the histogram. Only used if hwx.test was called with histobins set to a positive value.
...	other parameters passed to print.

---

whales.df *Bowhead whale data from Morin et al. 2012*

Description

Data from two populations of bowhead whales tested for 51 loci. This partial data set includes 279 individuals sampled from one population (P1) and 49 from another (P2). Each whale was genotyped at 51 loci, including SNP and microsattelite. These data reside in a single data frame object.

References

Morin et al., 2012

---

xcount *Find Exact Number of Genotype Tables*

Description

Use xcount to determine the exact number of tables (i.e., genotype numbers) for a given set of allele counts. This method enumerates all tables, and is best when the total number is less than 10^10 or so. This function is mostly called by hwx.test rather than directly by the user. If the number of tables is too large to enumerate with this method, use acount for an approximation.

Usage

```
xcount(m, safety = 1e+10, safeSecs = 10)
```

Arguments

m	vector containing the numbers of alleles of each type. Length must be at least 2 and all must be non-negative integers. It can also be a matrix of genotype counts.
safety	Stop execution if the approximate table number obtained from acount is more than this cutoff.
safeSecs	Time limit in seconds. Another safety feature to prevent getting stuck in a too-long computation

Value

The exact number of tables

References

The methods are described by Engels, 2009. **Genetics** 183:1431.

See Also

hwx.test, acount

Examples

```
# Allele counts from human Rh locus. Guo and Thompson, 1992, Figure 1
#
alleles <- c(15, 14, 11, 12, 2, 2, 1, 3)
xcount(alleles)
```

---

xcountCutoff *Determine immediately whether number of tables is over a limit*

Description

Calling scountCutoff gives you a quick answer to whether the number of tables is over a given cutoff. It is useful in deciding whether to analyze a data set with xtest or mtest. This function is used by hwx.test and not normally called directly by the user.

Usage

```
xcountCutoff(m, cutoff = 1e+07)
```

Arguments

m	vector containing the numbers of alleles of each type. It can also be a matrix of genotype counts, but not a vector of genotype counts.
cutoff	Is the number of tables above or below this value?

Value

TRUE or FALSE depending on whether the table count is above or below `cutoff`

Examples

```
#
alleles <- c(15, 14, 11, 12, 2, 2, 1, 3)
if(xcountCutoff(alleles)) cat("There are too many tables")
```

`xtest` *Performs an exact test with full enumeration for Hardy-Weinberg proportions.*

Description

Given a set of genotype counts, `xtest` examines all possible outcomes with the same set of allele counts. For each table, it computes four test statistics and compares them with the observed values. It returns the total probability of all tables with test statistics as "extreme" or more so than the observed. It can also plot a histogram of one of the statitistics if `histobins` is greater than zero. More about these four test statistics and other information can be found in the vignette. This function will not normally be called directly. Instead, `hwx.test` calls either `xtest` or `mtest` depending on which method is to be used.

Usage

```
xtest(c, statName = "LLR", histobins = 0, histobounds = c(0, 0),
  showCurve = T, safeSecs = 100, detail = 2)
```

Arguments

`c`	A matrix containing the genotype counts. It should be a square matrix, but only the lower-left half is used.
`statName`	can be "LLR", "Prob", "U", or "Chisq" depending on which one is to be ploted. Note that P values for all four are computed regardless of which one is specified with this parameter.
`histobins`	If 0 no histogram is plotted. If 1 or `TRUE` a histogram with 500 bins is plotted. If set to a number greater than 1, a histogram with `histobins` is plotted.
`histobounds`	A vector containing the left and right boundaries for the histogram's x axis. If you leave this as the default, `c(0,0)`, then `xtest` will compute reasonable bounds to include most of the distribution.
`showCurve`	whether to show a blue curve indicating the asymptotic (chi squared) distribution. This only works for `LLR` and `Chisq`
`safeSecs`	After this many seconds the calculation will be aborted. This is a safety valve to prevent attempts to compute impossibly large sets of tables.
`detail`	Determines how much detail is printed. If set to 0, nothing is printed (useful if you use `xtest` programmatically.).

Value

`xtest` returns a list components

`$ Pvalues`	The four computed P values corresponding to the test statistics: `LLR`, `Prob`, `U` and `Chisq` in that order.
`$ observed`	The four observed statistics in the same order as above
`$ tableCount`	The number of tables examined during the calculation
`$ ntrials`	placeholder
`$ genotypes`	The input matrix of genotype counts
`$ alleles`	The allele counts m corresponding to the input genotype counts
`$ statID`	Which test statistic was used if a histogram was plotted
`$ histobins`	If greater than zero, the number of bins to use for the histogram
`$ histobounds`	The lower and upper limits of the test statistic in the histogram
`$ histoData`	Vector of $histobins$ values for the histogram
`$ showCurve`	Whether the asymptotic curve should be plotted with the histogram

References

The methods are described by Engels, 2009. **Genetics** 183:1431.

See Also

`hwx.test`

Index