Package ‘chisquare’

November 3, 2023

Title  Chi-Square and G-Square Test of Independence, Residual Analysis, and Measures of Categorical Association

Version  0.7

Description  Provides the facility to perform the chi-square and G-square test of independence, calculates permutation-based p value, and provides measures of association such as Phi, odds ratio with 95 percent CI and p value, adjusted contingency coefficient, Cramer’s V and 95 percent CI, bias-corrected Cramer’s V, Cohen’s w, Goodman-Kruskal’s lambda, gamma and its p value, and tau, Cohen’s k and its 95 percent CI. It also calculates standardized, moment-corrected standardized, and adjusted standardized residuals, and their significance. Different outputs are returned in nicely formatted tables.

Depends  R (>= 4.0.0)

Imports  graphics (>= 4.2.0), gt (>= 0.3.1), stats (>= 4.2.0)

License  GPL (>= 2)

LazyData  true

RoxygenNote  7.2.3

NeedsCompilation  no

Author  Gianmarco Alberti [aut, cre]

Maintainer  Gianmarco Alberti <gianmarcoalberti@gmail.com>

Repository  CRAN

Date/Publication  2023-11-03 15:10:02 UTC

R topics documented:

  chisquare ......................................................... 2
diseases ............................................................ 13
  safety ............................................................. 13
social_class ......................................................... 14

Index  15
Description

The function performs the chi-square test (both in its original format and in the N-1 version) and the G-square test of independence on the input contingency table. It also calculates various measures of categorical association, returns standardized, moment-corrected standardized, and adjusted standardized residuals (with indication of their significance), and calculates relative and absolute contributions to the chi-square. The p value associated to the chi-square statistic is also calculated via a Monte Carlo method. The 95 percent confidence interval around the simulated p value is also calculated. Nicely-formatted output tables are rendered. Optionally, in 2xk tables (where k >= 2), a plot of the odds ratios can be rendered.

Visit this [LINK](#) to access the package’s vignette.

Usage

```r
chisquare(
  data,
  B = 999,
  plot.or = FALSE,
  reference.level = 1,
  row.level = 1,
  or.alpha = 0.05,
  adj.alpha = FALSE,
  format = "short",
  graph = FALSE,
  tfs = 14
)
```

Arguments

- **data**: Dataframe containing the input contingency table.
- **B**: Number of simulated tables to be used to calculate the Monte Carlo-based p value (999 by default).
- **plot.or**: Takes TRUE or FALSE (default) if the user wants a plot of the odds ratios to be rendered (only for 2xk tables, where k >= 2).
- **reference.level**: The index of the column reference level for odds ratio calculations (default: 1). The user must select the column level to serve as the reference level (only for 2xk tables, where k >= 2).
**row.level**  
The index of the row category to be used in odds ratio calculations (1 or 2; default: 1). The user must select the row level to which the calculation of the odds ratios make reference (only for 2xk tables, where k >= 2).

**or.alpha**  
The significance level used for the odds ratios’ confidence intervals (default: 0.05).

**adj.alpha**  
Takes TRUE or FALSE (default) if the user wants or does not want the significance level of the residuals (standarized, adjusted standardized, and moment-corrected) to be corrected using the Sidak’s adjustment method (see Details).

**format**  
Takes short (default) if the dataset is a dataframe storing a contingency table; if the input dataset is a dataframe storing two columns that list the levels of the two categorical variables, long will preliminarily cross-tabulate the levels of the categorical variable in the 1st column against the levels of the variable stored in the 2nd column.

**graph**  
Takes TRUE or FALSE (default) if the user wants or does not want to chart the distribution of the permuted chi-square statistic across the number of randomized tables set by the B parameter.

**tfs**  
Numerical value to set the size of the font used in the main body of the various output tables (14 by default).

**Details**

The function produces the following measures of categorical associations:

- Phi (with indication of the magnitude of the effect size; only for 2x2 tables)
- Phi corrected (with indication of the magnitude of the effect size; only for 2x2 tables)
- Phi signed (with indication of the magnitude of the effect size; only for 2x2 tables)
- Yule’s Q (only for 2x2 tables, includes p-value)
- Odds ratio (only for 2x2 tables, includes 95perc confidence interval, p value, and indication of the magnitude of the effect size)
- Adjusted contingency coefficient C (with indication of the magnitude of the effect size)
- Cramer’s V (with 95perc confidence interval; includes indication of the magnitude of the effect size)
- Bias-corrected Cramer’s V (with indication of the magnitude of the effect size)
- Cohen’s w (with indication of the magnitude of the effect size)
- W coefficient (includes 95perc confidence interval and magnitude of the effect size)
- Goodman-Kruskal’s lambda (both asymmetric and symmetric)
- Corrected version of lambda (both asymmetric and symmetric)
- Goodman-Kruskal’s tau (asymmetric) and gamma (with p-value)
- Cohen’s k (with 95perc confidence interval)

**Indication of the magnitude of the effect size for coefficients of association**

The function provides the effect size for the Phi, Phi corrected, Phi signed, Cadj, Cramer’s V, Cramer’s V bias-corrected, Cohen’s w, W, and for the Odds Ratio.
With the exception of the latter two (for which see further down), the effect size for the other measures of association is based on Cohen 1988. For a useful summary of the approach used, refer to Sheskin 2011, 675 and following pages.

While Phi, Phi corrected, Phi signed, and w are assessed against the well-known Cohen’s classification scheme (small 0.1, medium 0.3, large 0.5), Cadj and the two versions of V are preliminary turned into Cohen’s w before attaching either the "small effect", or "medium effect", or "large effect" label.

For instance, using the in-built ‘social_class’ 3x4 cross-tab, Cramer’s V is 0.233. Before attaching an effect size label, V is turned into w by multiplying it by the square root of the number of rows or columns, whichever is smaller, minus 1. Therefore, the value of V that has to be given a label would translate into 0.233*sqrt(3-1)=0.33. The latter, according to Cohen’s guidelines, can be labelled as pointing to a "medium effect". The bias-corrected version of V is 0.21, which would translate into 0.21*sqrt(3-1)=0.296 which, in turn, can be labelled as indicating a "small effect".

It is important to note that, when working with V, the transformation into w involves (as said) multiplying the V value by the square root of the lesser between the number of rows or columns in the table, minus one. This number essentially represents the dimensionality of the table.

Consider a V value of 0.35. The way it translates to w, and subsequently its effect size interpretation, can change based on the table’s dimensionality:
for a 2x2 table: w=0.35*sqrt(2-1)=0.35, indicating a "medium" effect;
for a 3x3 table: w=0.35*sqrt(2-1)=0.49, still falling under the "medium" effect category;
for a 4x4 table: w=0.35*sqrt(4-1)=0.61, which now is interpreted as a "large" effect.

The examples illustrate that the larger the table dimensionality (i.e., more rows or columns), the larger the ’w’ value will be for a given V. This means that for the same V value, the interpreted effect size can shift from "medium" in a smaller table to "large" in a larger table. It is crucial to be aware of this as it highlights that the same V value can imply different magnitudes of effect depending on the table’s dimensionality.

In simpler terms, the threshold for determining a "large" effect, for instance, becomes more accessible to reach as the table’s size increases.

Cadj is turned into w as follows: sqrt(Cadj^2/(1-Cadj^2)).

For more details on the entire approach, refer to Sheskin 2011 cited above.

**Monte Carlo p-value for the chi-square statistic**

The p-value of the observed chi-square statistic is also calculated on the basis of a Monte Carlo (permutation-based) approach, using B random tables created under the Null Hypothesis of independence.

For the rationale of this approach, see for instance the description in Beh-Lombardo 2014: 62-64. The permutation-based p value is calculated as follows:
(1 + sum(chistat.perm > chisq.stat))/(1 + B), where

chistat.perm is a vector storing the B chi-square statistics generated under the Null Hypothesis, and chisq.stat is the observed chi-square statistic.

For the logic of the calculation, see for example Baddeley et al. 2016: 387.

**Confidence interval around the Monte Carlo p-value**

The function calculates the 95 percent Confidence Interval around the Monte Carlo p-value. The Wald CI quantifies the uncertainty around the Monte Carlo p-value estimate. For a 95 percent CI, the standard z-value of 1.96 is used. The standard error for the estimated p-value is computed as the square root of (estimated p-value * (1 - estimated p-value) / number of simulations).

The lower and upper bounds of the CI are then calculated as follows:
- Lower Confidence Interval = estimated p-value - (z-value * standard error)
- Upper Confidence Interval = estimated p-value + (z-value * standard error)

Finally, the lower and upper CIs are clipped to lie within 0 and 1.

**Chi-square statistics adjusted using the (N-1)/N correction**

The correction is done by multiplying the chi-square statistics by (N-1)/N, where N is the table grand total (sample size). The p-value of the corrected statistic is calculated the regular way. The correction seems particularly relevant for tables where N is smaller than 20 and where the expected frequencies are equal or larger than 1. The corrected chi-square test proves more conservative when the sample size is small. As N increases, the term (N-1)/N approaches 1, making the adjusted chi-square value virtually equivalent to the unadjusted value.

See: Upton 1982; Rhoades-Overall1982; Campbell 2007; Richardson 2011.

**Moment-corrected standardized residuals**

The moment-corrected standardized residuals are calculated as follows:

\[ \text{stand.res} / \sqrt{(nr - 1) * (nc - 1)/(nr * nc)} \], where

stand.res is each cell’s standardized residual, nr and nc are the number of rows and columns respectively.


**Adjusted standardized residuals**

The adjusted standardized residuals are calculated as follows:

\[ \text{stand.res}[i, j] / \sqrt{(1 - sr[i]/n) * (1 - sc[j]/n)} \], where

stand.res is the standardized residual for cell \( ij \), sr is the row sum for row \( i \), sc is the column sum for column \( j \), and \( n \) is the table grand total. The adjusted standardized residuals should be used in place of the standardised residuals since the latter are not truly standardised because they have a nonunit variance. The standardised residuals therefore underestimate the divergence between the observed and the expected counts. The adjusted standardized residuals (and the moment-corrected ones) correct that deficiency.
Significance of the residuals
The significance of the residuals (standardized, moment-corrected standardized, and adjusted stan-
dardized) is assessed using alpha 0.05 or, optionally (by setting the parameter ‘adj.alpha’ to TRUE),
using an adjusted alpha calculated using the Sidak’s method:

\[ \text{alpha.adj} = 1 - (1 - 0.05) \left( \frac{1}{\text{nr} \times \text{nc}} \right) \]

\( \text{nr} \) and \( \text{nc} \) are the number of rows and columns in the table respectively. The adjusted alpha is then
converted into a critical two-tailed \( z \) value.

Cells’ relative contribution (in percent) to the chi-square statistic
The cells’ relative contribution (in percent) to the chi-square statistic is calculated as:

\[ \frac{\text{chisq.values}}{\text{chisq.stat}} \times 100, \text{ where} \]

\( \text{chisq.values} \) and \( \text{chisq.stat} \) are the chi-square value in each individual cell of the table and the value
of the chi-square statistic, respectively. The average contribution is calculated as \( 100/(\text{nr} \times \text{nc}) \),
where \( \text{nr} \) and \( \text{nc} \) are the number of rows and columns in the table respectively.

Cells’ absolute contribution (in percent) to the chi-square statistic
The cells’ absolute contribution (in percent) to the chi-square statistic is calculated as:

\[ \frac{\text{chisq.values}}{\text{n}} \times 100, \text{ where} \]

\( \text{chisq.values} \) and \( \text{n} \) are the chi-square value in each individual cell of the table and the table’s grant
total, respectively. The average contribution is calculated as sum of all the absolute contributions
divided by the number of cells in the table.
For both the relative and absolute contributions to the chi-square, see: Beasley-Schumacker 1995: 90.

Phi corrected
To further refine Phi, a corrected version has been introduced. It accounts for the fact that the origi-
nal coefficient (1) might not reach its maximum value of 1 even when there is a perfect association
between the variables, and (2) it is not directly comparable across tables with different marginals.
To calculate Phi-corrected, one first computes Phi-max, which represents the maximum possible
value of Phi under the given marginal totals. Phi-corrected is equal to Phi/Phi-max.
For more details see: Cureton 1959; Liu 1980; Davenport et al. 1991; Rash et al. 2011.

95perc confidence interval around Cramer’s V
The calculation of the 95perc confidence interval around Cramer’s V is based on Smithson 2003: 39-
41, and builds on the R code made available by the author on the web (http://www.michaelsmithson.online/stats/CIstuff/CI.html)
**Bias-corrected Cramer’s V**
The bias-corrected Cramer’s V is based on Bergsma 2013: 323–328.

**W coefficient**
It addresses some limitations of Cramer’s V. When the marginal probabilities are unevenly distributed, V may overstate the strength of the association, proving pretty high even when the overall association is weak. W is based on the distance between observed and expected frequencies. It uses the squared distance to adjust for the unevenness of the marginal distributions in the table. The indication of the magnitude of the effect size is based (with minor changes in terminology) on Kvalseth 2018a. Unlike Kvalseth 2018a, the calculation of the 95 percent confidence interval is based on a bootstrap approach (employing 10k resampled tables, and the 2.5th and 97.5th percentiles of the bootstrap distribution).

For more details see: Kvalseth 2018a.

**Corrected Goodman-Kruskal’s lambda**
The corrected Goodman-Kruskal’s lambda adeptly addresses skewed or unbalanced marginal probabilities which create problems to the traditional lambda. By emphasizing categories with higher probabilities through a process of squaring maximum probabilities and normalizing with marginal probabilities, this refined coefficient addresses inherent limitations of lambda.

For more details see: Kvalseth 2018b.

**Odds Ratio**
The odds ratio is calculated for 2x2 tables. In case of zeros along any of the table’s diagonal, the Haldane-Anscombe correction is applied. It consists in adding 0.5 to every cell of the table before calculating the odds ratio. For tables of size 2xk (where k >= 2), pairwise odds ratios can be plotted (along with their confidence interval) by setting the or.alpha parameter to TRUE. The mentioned correction is also applied to the calculation of those pairwise odds ratios (for more information on the plot, see further below).


**Odds Ratio effect size magnitude**
The magnitude of the effect indicated by the odds ratio is based on the thresholds (and corresponding reciprocal) suggested by Chen et al 2010:

- OR < 1.68 - Very small
- 1.68 <= OR < 3.47 - Small
- 3.47 <= OR < 6.71 - Medium
- OR >= 6.71 - Large

**Odd Ratios plot**
For 2xk table, where k >= 2:
by setting the plor.or parameter to TRUE, a plot showing the odds ratios and their 95percent confidence interval will be rendered. The confidence level can be modified via the or.alpha parameter. The odds ratios are calculated for the column levels, and one of them is to be selected by the user as a reference for comparison via the reference.level parameter (set to 1 by default). Also, the user
may want to select the row category to which the calculation of the odds ratios makes reference (using the row.level parameter, which is set to 1 by default). If any of the pairwisely-generated 2x2 tables on which the odds ratio is calculated features zeros along any of the diagonal, the Haldane-Anscombe correction is applied (see above).

To better understand the rationale of plotting the odds ratios, consider the following example, which uses on the famous Titanic data:

Create a 2x3 contingency table:
```r
c CreateTable <- matrix(c(123, 158, 528, 200, 119, 181), nrow = 2, byrow = TRUE)
colnames(CreateTable) <- c("1st", "2nd", "3rd")
rownames(CreateTable) <- c("Died", "Survived")
```

Now, we perform the test and visualise the odds ratios:
```r
chisquare(CreateTable, plot.or=TRUE, reference.level=1, row.level=1)
```

In the rendered plot, we can see the odds ratios and confidence intervals for the second and third column level (i.e., 2nd class and 3rd class) because the first column level has been selected as reference level. The odds ratios are calculated making reference to the first row category (i.e., Died). From the plot, we can see that, compared to the 1st class, passengers on the 2nd class have 2.16 times larger odds of dying; passengers on the 3rd class have 4.74 times larger odds of dying compared to the 1st class.

Note that if we set the row.level parameter to 2, we make reference to the second row category, i.e. Survived:
```r
chisquare(CreateTable, plot.or=TRUE, reference.level=1, row.level=2)
```

In the plot, we can see that passengers in the 2nd class have 0.46 times the odds of surviving of passengers in the 1st class, while passengers from the 3rd class have 0.21 times the odds of surviving of those travelling in the 1st class.

Other measures of categorical association
For the other measures of categorical association provided by the function, see for example Sheskin 2011: 1415-1427.

Additional notes on calculations:
- the Phi coefficient is based on the chi-square statistic as per Sheskin 2011’s equation 16.21, whereas the Phi signed is after Sheskin’s equation 16.20;
- the 2-sided p value of Yule’s Q is calculated following Sheskin 2011’s equation 16.24;
- Cohen’s w is calculated as \[ V \times \sqrt{\min(nr, nc) - 1} \], where \( V \) is Cramer’s \( V \), and \( nr \) and \( nc \) are the number of rows and columns respectively; see Sheskin 2011: 679;
- the 2-tailed p value of Goodman-Kruskal’s gamma is based on the associated z-score calculated as per Sheskin 2011’s equation 32.2;
- the symmetric version of Goodman-Kruskal’s lambda is calculated as per Reynolds 1984: 55-57;
- **Goodman-Kruskal’s tau** is calculated as per Reynolds 1984: 57-60;
- **Cohen’s k** is calculated as per Sheskin 2011: 688-689 (equation 16.30).

**Value**

The function produces **optional charts** (distribution of the permuted chi-square statistic and a plot of the odds ratios between a reference column level and the other ones, the latter only for 2xk tables where \( k \geq 2 \)), and a number of **output tables** that are nicely formatted with the help of the *gt* package. The output tables are listed below:

- Input contingency table (with some essential analytical results annotated at the bottom)
- Expected frequencies
- Cells’ chi-square value
- Cells’ relative contribution (in percent) to the chi-square statistic (cells in RED feature a larger-than-average contribution)
- Cells’ absolute contribution (in percent) to the chi-square statistic (colour same as above)
- Standardized residuals (RED for large significant residuals, BLUE for small significant residuals)
- Moment-corrected standardized residuals (colour same as above)
- Adjusted standardized residuals (colour same as above)
- Table of output statistics, p values, and association measures

Also, the function returns a **list containing the following elements**:

- **input.table**:  
  - *crosstab*: input contingency table.

- **chi.sq.related.results**:  
  - *exp.freq*: table of expected frequencies.
  - *chisq.values*: cells’ chi-square value.
  - *chisq.relat.contrib*: cells’ relative contribution (in percent) to the chi-square statistic.
  - *chisq.abs.contrib*: cells’ absolute contribution (in percent) to the chi-square statistic.
  - *chisq.statistic*: observed chi-square value.
  - *chisq.p.value*: p value of the chi-square statistic.
  - *chisq.p.value.MC CI lower boundary*: lower boundary of the 95 percent CI around the Monte Carlo p value.
  - *chisq.p.value.MC CI upper boundary*: upper boundary of the 95 percent CI around the Monte Carlo p value.
  - *chisq.adj*: chi-square statistic adjusted using the (N-1)/N correction.

- **G.square**:  
  - *Gsq.statistic*: observed G-square value.
• residuals:

• chi.sq.based.assoc.measures:
  – Phi: Phi coefficient (only for 2x2 tables).
  – Phi corr: corrected Phi coefficient (only for 2x2 tables).
  – Phi signed: signed Phi coefficient (only for 2x2 tables).
  – Cadj: adjusted contingency coefficient C.
  – Cramer’s V: Cramer’s V coefficient.
  – Cramer’s V CI lower boundary: lower boundary of the 95perc CI.
  – Cramer’s V CI upper boundary: upper boundary of the 95perc CI.
  – Cramer’s Vbc: bias-corrected Cramer’s V coefficient.
  – w: Cohen’s w.
  – W CI lower boundary: lower boundary of the 95perc CI.
  – W CI upper boundary: upper boundary of the 95perc CI.

• non.chi.sq.based.assoc.measures:
  – Yule’s Q: Q coefficient (only for 2x2 tables).
  – Yule’s Q p.value: 2-tailed p value of Yule’s Q.
  – Odds ratio: odds ratio (only for 2x2 tables).
  – Odds ratio CI lower boundary: lower boundary of the 95perc CI.
  – Odds ratio CI upper boundary: upper boundary of the 95perc CI.
  – Odds ratio p.value: p value of the odds ratio.
  – lambda (rows dep.): Goodman-Kruskal’s lambda coefficient (considering the rows being the dependent variable).
  – lambda (cols dep.): Goodman-Kruskal’s lambda coefficient (considering the columns being the dependent variable).
  – lambda (symmetric): Goodman-Kruskal’s symmetric lambda coefficient.
  – lambda corrected (rows dep.): corrected version of the lambda coefficient (considering the rows being the dependent variable).
  – lambda corrected (cols dep.): corrected version of the lambda coefficient (considering the columns being the dependent variable).
  – lambda corrected (symmetric): corrected version of the symmetric lambda coefficient.
  – tau (rows dep.): Goodman-Kruskal’s tau coefficient (considering the rows being the dependent variable).
  – tau (cols dep.): Goodman-Kruskal’s tau coefficient (considering the columns being the dependent variable).
  – gamma: Goodman-Kruskal’s gamma coefficient.
  – gamma.p.value: 2-sided p value for the Goodman-Kruskal’s gamma coefficient.
  – k: Cohen’s k.
  – k CI lower boundary: lower boundary of the 95perc CI.
Note that the p-values returned in the above list are expressed in scientific notation, whereas the ones reported in the output table featuring the tests’ result and measures of association are reported as broken down into classes (e.g., <0.05, or <0.01, etc), with the exception of the Monte Carlo p-value and its CI.

The following examples, which use in-built datasets, can be run to familiarise with the function:

- perform the test on the in-built ’social_class’ dataset
  result <- chisquare(social_class)

- perform the test on a 2x2 subset of the ’diseases’ dataset
  mytable <- diseases[3:4,1:2]
  result <- chisquare(mytable)

- perform the test on a 2x2 subset of the ’safety’ dataset
  mytable <- safety[c(4,1),c(1,6)]
  result <- chisquare(mytable)

- build a toy dataset in ’long’ format (gender vs. opinion about death sentence)
  mytable <- data.frame(GENDER=c(rep("F", 360), rep("M", 340)),
                        OPINION=c(rep("oppose", 235),
                                  rep("favour", 125),
                                  rep("oppose", 160),
                                  rep("favour", 180)))

- perform the test specifying that the input table is in ’long’ format
  result <- chisquare(mytable, format="long")

References

Examples

# Perform the test on the in-built 'social_class' dataset
result <- chisquare(social_class, B=99)

# Perform the test on a 2x2 subset
result <- chisquare(social_class[c(1:2), c(1:2)], B=99)
<table>
<thead>
<tr>
<th>diseases</th>
<th>Dataset: Cross-tabulation of quantity of tobacco smoked daily vs. cause of death</th>
</tr>
</thead>
</table>

**Description**

Cross-tabulation (15x4) of the amount of tobacco smoked on a daily basis (in gramms) against cause of death.

**Usage**

data(diseases)

**Format**

dataframe

<table>
<thead>
<tr>
<th>safety</th>
<th>Dataset: Cross-tabulation of people’s feeling of safety vs. town size</th>
</tr>
</thead>
</table>

**Description**

Cross-tabulation (4x6).

**Usage**

data(safety)

**Format**

dataframe
social_class

| social_class | Dataset: Cross-tabulation of social class vs. diagnostic category for a sample of psychiatric patients |

Description


Usage

data(social_class)

Format

dataframe
Index

* chiperm
  chisquare, 2
* datasets
  diseases, 13
  safety, 13
  social_class, 14

chisquare, 2

diseases, 13

safety, 13

social_class, 14