G Goodness-of-Fit Test (test_g_gof)

a.k.a. Wilks or Likelihood Ratio Goodness-of-Fit

Author: P. Stikker
Website: https://peterStatistics.com
YouTube: https://www.youtube.com/stikpet
Version: 0.1 (2023-01-07)

Introduction

The *test_g_gof* function (and *test_g_gof_arr* in VBA) performs a G Goodness-of-Fit test. The test could be used to compare the proportions from different categories. The null-hypothesis is roughly that the proportions are all the same. If the p-value is too small (usually below 0.05) the assumption is rejected, indicating that at least two categories will have a different proportion in the population.

This document contains the details on how to use the functions, and formulas used in them.

1 About the Function

1.1 Input parameters:

- **data**
  The data to be used. Note for Python this needs to be a pandas data series.

- *Optional parameters*
  - **expCount** (default is none)
    A table with two columns. One with the categories and another with the expected counts. In Pandas this needs to be a dataframe.

  - **cc** (default is none)
    which (if any) continuity correction to use. Either
    - “none”: no correction
    - “yates”: Yates
    - “pearson”: E.S. Pearson
    - “williams”: Williams

  - **out** (default is “pvalue”) – only applies to VBA *test_g_gof*
    Choice what to show as result. Either:
    - “pvalue”: show the p-value (significance)
    - “df”: the degrees of freedom
    - “statistic”: show the test-statistic used

1.2 Output:

- The **test-statistic (chi-square value)**, **degrees of freedom**, **p-value** and **test** used. Except for the non-array version in VBA (Excel) which will only show the requested Alternative Ratio.

- The array version in VBA (*test_g_gof_arr*) requires **two rows** and **four columns**.

1.3 Dependencies

- **Excel**
  None.
  You can run the **test_g_gof_addHelp** macro so that the function will be available with some help in the 'User Defined' category in the functions overview.

- **Python**
  The following additional libraries will have to be installed/loaded:
  - *pandas*
    the data input needs to be a pandas data series, and the output is also a pandas dataframe.

  - *math*
    the *log* function from Python's math library is needed

- **R**
  No other libraries required.

2 Examples

2.1 Excel

	A	B	C	D	E	F	G	H	I	J
1	Marital		Expected counts							
2	MARRIED		MARRIED	5						
3	DIVORCED		DIVORCED	5						
4	MARRIED		NEVER MARRIED	5						
5	SEPARATED		SEPARATED	5						
6	DIVORCED									
7	NEVER MARRIED				E10 =	=ts_g_gof($A$2:$A$20;;$D10;E$9)				
8	DIVORCED									
9	DIVORCED			cc	statistic	df	pvalue			
10	NEVER MARRIED			none	3,397304	3	0,334328			
11	MARRIED			yates	1,994312	3	0,573588			
12	MARRIED			pearson	3,218498	3	0,359148			
13	MARRIED			williams	3,25456	3	0,354017			
14	SEPARATED									
15	DIVORCED			0,334328	=ts_g_gof(A2:A20;C2:D5)					
16	NEVER MARRIED									
17	NEVER MARRIED			statistic	df	p-value	test			
18	DIVORCED			3,397304	3	0,334328	G (Likelihood Ratio) test of goodness-of-fit			
19	DIVORCED									
20	MARRIED			D12:G13	=ts_g_gof_arr(A2:A20)					

2.2 Python

```
[2]: #Example
data = pd.DataFrame(["MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED",
                     "NEVER MARRIED", "DIVORCED", "DIVORCED", "NEVER MARRIED",
                     "MARRIED", "MARRIED", "MARRIED", "SEPARATED", "DIVORCED",
                     "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED"],
                    columns=["marital"])
```

```
[3]: ts_g_gof(data)
```

[3]:

	statistic	df	p-value	test
0	3.397304	3	0.334328	G test of goodness-of-fit

```
[4]: ts_g_gof(data, cc="pearson")
```

[4]:

	statistic	df	p-value	test
0	3.218498	3	0.359148	G test of goodness-of-fit, with E. Pearson continuity correction

```
[5]: eCounts = pd.DataFrame({'category' : ["MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"], 'count' : [5,5,5,5]})
ts_g_gof(data, eCounts)
```

[5]:

	statistic	df	p-value	test
0	3.397304	3	0.334328	G test of goodness-of-fit

2.3 R

```
>
> data <- c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
+           "DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED",
+           "SEPARATED", "DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
> eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
> ts_g_gof(data)
    chiVal df      pVal                             testUsed
1 3.397304  3 0.3343277 Pearson chi-square test of goodness-of-fit
> ts_g_gof(data, cc="yates")
    chiVal df      pVal                                                                  testUsed
1 1.994312  3 0.5735881 Pearson chi-square test of goodness-of-fit , with Yates continuity correction
> ts_g_gof(data, cc="pearson")
    chiVal df      pVal                                                                       testUsed
1 3.218498  3 0.3591482 Pearson chi-square test of goodness-of-fit , with E. Pearson continuity correction
> ts_g_gof(data, cc="williams")
   chiVal df      pVal                                                                     testUsed
1 3.25456  3 0.3540173 Pearson chi-square test of goodness-of-fit , with Williams continuity correction
> ts_g_gof(data, eCounts)
    chiVal df      pVal                             testUsed
1 3.397304  3 0.3343277 Pearson chi-square test of goodness-of-fit
> |
```

3 Details of Calculations

3.1 The Original Test

The G Goodness-of-Fit test uses:

$$\chi^2_{LR} = 2 \times \sum_{i=1}^{k} F_i \times \ln\left(\frac{F_i}{E_i}\right)$$

$$df = k - 1$$

$$sig. = 1 - \chi^2(\chi^2_{LR}, df)$$

If the expectation about the population, is that all categories have the same frequency, then:

$$E_i = \frac{n}{k}$$

$$n = \sum_{i=1}^{k} F_i$$

Symbols used:

- k the number of categories
- F_i the (absolute) frequency of category i
- E_i the expected frequency of category i
- n the sample size, i.e. the sum of all frequencies
- $\chi^2(\ldots)$ the chi-square cumulative density function
- $\ln(\ldots)$ the natural logarithm function

3.2 Yates Continuity Correction

This correction is usually only recommended if the degrees of freedom is two. For a goodness-of-fit test this means only if you have two categories.

$$F_i' = \begin{cases} F_i - 0.5 & if\ F_i > E_i \\ F_i + 0.5 & if\ F_i < E_i \\ F_i & if\ F_i = E_i \end{cases}$$

$$\chi^2_{LR-Yates} = 2 \times \sum_{i=1}^{k} \left(F_i' \times ln\left(\frac{F_i'}{E_i}\right)\right)$$

If $F_i' = 0$, then $F_i' \times ln\left(\frac{F_i'}{E_i}\right) = 0$

3.3 E.S. Pearson correction

$$\chi^2_{LR-EP.GoF} = \frac{n-1}{n} \times \chi^2_{LR}$$

3.4 Williams correction

$$\chi^2_{LR-W.GoF} = \frac{\chi^2_{LR}}{q}$$

With:

$$q = 1 + \frac{k^2 - 1}{6 \times n \times df}$$

If $df = k - 1$ (which usually is the case with a GoF test, except if you have an intrinsic null hypothesis), the formula can be simplified to:

$$q = 1 + \frac{k + 1}{6 \times n}$$

4 Sources

Wilks describes a theorem in his paper:

Therefore, except for terms of order $1/\sqrt{n}$,

(9) $$-2 \log \lambda = \chi_0^2 .$$

(Wilks, 1938, p. 62)

By setting lambda accordingly the equation can be found.

The term 'Likelihood Ratio Goodness-of-Fit' can for example be found in an article from Quine and Robinson (1985), the term 'Wilks's likelihood ratio test' can also be found in Li and Babu (2019, p. 331), while the term G-test is found in Hoey (2012, p. 4)

Interestingly, the likelihood ratio can be more formally related to the χ^2 test, by considering the G-test, defined as [5]

$$G = 2 \sum_i O_i \log(O_i/E_i)$$

(Hoey, 2012, p. 4)

The Pearson correction is found as:

and $m+n=N$.* It is seen that the ratio d/s_d is identical with the ratio u of equation (22), except for a factor $\sqrt{[(N-1)/N]}$ which is unimportant in large samples. Thus the classical test is practically identical with that suggested in paras. 40–42 above, though the two tests are differently derived.

(Pearson, 1947, p. 157)

The Williams correction is from Williams (1976)

$q = 1 + \frac{1}{6\nu n}$(sum of reciprocals of expected cell frequencies

− sums of expectations of marginal frequencies in the numerators of the maximum likelihood estimators

+ sums of expectations of marginal frequencies in the denominators of the maximum likelihood estimators).

In general q is a function of the expected frequencies. To determine a numerical value for q these expected frequencies must in practice be replaced by their maximum likelihood estimators.

A much easier alternative is to use the minimum value q_{min} of q given by

$$q_{min} = 1 + \phi(a^2, b^2, \ldots)/(6\nu n),$$

where $\phi(a, b, \ldots)$ is the deviance degrees of freedom ν expressed as a function of the factor levels $a, b, \ldots$. The difference between q and q_{min} will often be small, and the use of $q = q_{min}$

(Williams, 1976, p. 36)

The formula used is adopted from McDonald (2014).

References

Hoey, J. (2012). *The two-way likelihood ratio (G) test and comparison to two-way chi squared test*. 1–6. https://doi.org/10.48550/ARXIV.1206.4881

Li, B., & Babu, G. J. (2019). *A graduate course on statistical inference*. Springer.

McDonald, J. H. (2014, December). *Small numbers in chi-square and G–tests*. Handbook of Biological Statistics. http://www.biostathandbook.com/small.html

Pearson, E. S. (1947). The choice of statistical tests illustrated on the Interpretation of data classed in a 2 × 2 table. *Biometrika*, *34*(1/2), 139–167. https://doi.org/10.2307/2332518

Quine, M. P., & Robinson, J. (1985). Efficiencies of chi-square and likelihood Ratio goodness-of-fit tests. *The Annals of Statistics*, *13*(2), 727–742. https://doi.org/10.1214/aos/1176349550

Wilks, S. S. (1938). The large-sample distribution of the likelihood ratio for testing composite hypotheses. *The Annals of Mathematical Statistics*, *9*(1), 60–62. https://doi.org/10.1214/aoms/1177732360

Williams, D. A. (1976). Improved likelihood ratio tests for complete contingency tables. *Biometrika*, *63*(1), 33–37. https://doi.org/10.2307/2335081