# Package ‘stikpetR’

February 23, 2025

**Title** Functions for Analysing a Survey

**Version** 0.0.3.0

**Description** The goal of stikpetR is to provide functions for statistical analysis of surveys. The functions are documented with the formulas and references used. The functions are NOT optimized but instead relatively easy to follow. If you would look at the source code it is most often a combination of if statements and for loops..

**Depends** stats,
graphics

**Imports** gsl,
fMultivar,
polycor,
pspearman,
ggplot2,
lattice

**License** GPL (>= 3)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.0

**Suggests** knitr,
rmarkdown

**VignetteBuilder** knitr

## R topics documented:

---

## `di_kendall_tau` *Kendall Tau Distribution*

### Description

Kendall Tau Distribution

### Usage

```
di_kendall_tau(n, tau, method = c("kendall", "AS71"))
```

### Arguments

| | |
|---|---|
| `n` | the sample size (number of pairs) |
| `tau` | Kendall tau value |
| `method` | algorithm to use |

### Details

If *method="AS71"* Algorithm AS 71 (Best & Gipps, 1974) will be used, by running the helper function *he_AS71(S, n)*. The test statistic is:

$$S = \binom{n}{2} \times |\tau| = \frac{n \times (n-1)}{2} \times |\tau|$$

AS 71 returns upper values only, so they get doubled for a two-sided test.

If *method="kendall"* the algorithm found at https://github.com/scipy/scipy/blob/v1.10.1/scipy/stats/_mstats_basic.py#L7 L898 was adapted. This refers to Kendall (1970), and uses the helper function *he_kendall(n, C)*. Where $C = n_c$, i.e. the number of concordant pairs. This algorithm already returns a two-tailed result.

### Value

| | |
|---|---|
| `pValue` | the two-tailed significance (p-value) |

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

Best, D. J., & Gipps, P. G. (1974). Algorithm AS 71: The upper tail probabilities of Kendall's tau. *Applied Statistics, 23*(1), 98–100. https://doi.org/10.2307/2347062

---

## `di_mcdf` *Multinomial Cumulative Distribution Function*

### Description

This is a function for the cumulative multinomial probability. It returns the probability of a distribution as given in F for a sample size of sum of F, where the probability for each category is given as in P, or a distribution even more rare. It is a generalization of the binomial distribution.

The distribution is also described at PeterStatistics.com

### Usage

```
di_mcdf(F, P, method = "loggamma")
```

### Arguments

| | |
|---|---|
| `F` | list with the observed counts |
| `P` | list with the probabilities for each category |
| `method` | optional the calculation method to use. Either "loggamma" (default), "factorial", "gamma", "mprob". |

## Details

The function first determines all possible arrangements over k categories that sum to n, using the **find_combinations()** function. It then uses the **di_pmf()** function to determine the probability for each of these, and sums those that are less or equal to the sample version.

This distribution is used in a Multinomial Goodness-of-Fit Test. The stikpetR library has a function `ts_multinomial_gof` for this, but it uses the dmultinomial function from R.

## Value

A float with the requested probability

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
freq = c(3, 6, 2, 9)
prob = c(0.2, 0.3, 0.1, 0.4)
di_mcdf(freq, prob)
```

---

# `di_mpmf` *Multinomial Probability Mass Function*

## Description

This is a function for the multinomial probability. It returns the probability of a distribution as given in F for a sample size of sum of F, where the probability for each category is given as in P. It is a generalization of the binomial distribution.

The distribution is also described at PeterStatistics.com

## Usage

```
di_mpmf(F, P, method = "loggamma")
```

## Arguments

| | |
|---|---|
| `F` | list with the observed counts |
| `P` | list with the probabilities for each category |
| `method` | optional the calculation method to use. Either "loggamma" (default), "factorial", "gamma", "mprob". |

## Details

If *method=factorial* the following formula is used:

$$mpmf\left(F, P\right) = \frac{n!}{\prod_{i=1}^{k}\left(F_i!\right)} \times \prod_{i=1}^{k} P_i^{F_i}$$

This formula was most likely already used by for example Edgeworth (1905), but can for example also be found in Berry and Mielke (1995, p. 769)

If *method=gamma*:

$$mpmf\left(F, P\right) = \frac{\Gamma\left(1+n\right)}{\prod_{i=1}^{n}\Gamma\left(1+F_i\right)} \times \prod_{i=1}^{k} P_i^{F_i}$$

If *method=loggamma*:

$$mpmf\left(F, P\right) = e^{\ln\left(mpmf\left(F,P\right)\right)}$$

$$\ln\left(mpmf\left(F, P\right)\right) = \ln\left(\Gamma\left(n+1\right)\right) + \sum_{i=1}^{k} F_i \times \ln\left(P_i\right) - \ln\left(\Gamma\left(F_i+1\right)\right)$$

This formula can for example be found in Arnold (2018).

If *method=mprob* the algorithm from García-Pérez (1999) is used:

1. Determine $F^*$, the counts in descending order, and move the elements in $P$ accordingly creating $P^*$.
2. Set $pmf = 1$, $t = P_1^*$, $i = 2$, $x = 0$, and $m = F_1^*$
3. Set $l = F_i^*$. For $r = 1$ to $l$ do:
    - update $x = x + 1$
    - if $x > F_1^*$ then set $t = 1$ (else nothing)
    - update $pmf = pmf \times t \times P_i^* \times \frac{r+m}{r}$
4. If $i = k$, then go to step 5, otherwise update $i = i + 1$, $m = m + F_i^*$ and go to step 3
5. If $x < F_1^*$ then for $r = x + 1$ to $F_1^*$ update $pmf = pmf \times P_1^*$

This distribution is used in a Multinomial Goodness-of-Fit Test. The stikpetR library has a function `ts_multinomial_gof` for this, but it uses the dmultinomial function from R.

## Value

A float with the requested probability

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Arnold, J. (2018, December 3). Maximum Likelihood for the multinomial distribution (bag of words) [Blog]. Jakuba. https://blog.jakuba.net/maximum-likelihood-for-multinomial-distribution/

Berry, K. J., & Mielke, P. W. (1995). Exact cumulative probabilities for the multinomial distribution. *Educational and Psychological Measurement, 55*(5), 769–772. doi:10.1177/0013164495055005008

Edgeworth, F. Y. (1905). The law of error. *Transactions of the Cambridge Philosophical Society, 20*, 36–66.

García-Pérez, M. A. (1999). MPROB: Computation of multinomial probabilities. *Behavior Research Methods, Instruments, & Computers, 31*(4), 701–705. doi:10.3758/BF03200749

## Examples

```
freq = c(3, 6, 2, 9)
prob = c(0.2, 0.3, 0.1, 0.4)
di_mpmf(freq, prob)
```

---

# `di_mwwcdf` *Mann-Whitney-Wilcoxon Cumulative Distribution Function*

## Description

This function returns the cumulative probability for the specified U statistic, given n1 and n2 cases in each category.

It first uses the di_mwwd function to determine the distribution up to the specified u value, sums these results and divides it by the total number of possible arrangements.

The pwilcox() function from R's stats library does the same, and is probably more optimized than this function.

## Usage

```
di_mwwcdf(u, n1, n2)
```

## Arguments

| | |
|---|---|
| `u` | int, the U test statistic |
| `n1` | int, the sample size of the first category |
| `n2` | int the sample size of the second category |

## Details

See the details in di_mwwd() on how the frequency distribution is determined. The sum of these is then divided by the total number of possibilities, which is the number of ways we can choose $n_1$ items out of $n$, without replacement. This is the binomial coefficient, or number of combinations:

$$C(n, n_1) = nCr(n, n_1) = \binom{n}{n_1} = \frac{n!}{n_1! \times (n - n_1)!}$$

To convert a W statistic to a U statistic use:

$$U = W - \frac{n_1 \times (n_1 + 1)}{2}$$

## Value

p : the cumulative probability

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

## `di_mwwd` *Mann-Whitney-Wilcoxon Distribution*

---

### Description

This distribution is also referred to as a permutation distribution.

It is used in the Mann-Whitney U and Wilcoxon Rank Sum test.

In this version the U-statistic is used as input, and the sample sizes of each of the two categories. This function will return the counts (frequency) of each possible U value from 0 till the provided u value. If all possible values need to be shown, simply set u = i*j.

### Usage

```
di_mwwd(u, n1, n2)
```

### Arguments

| | |
|---|---|
| `u` | int, the U test statistic |
| `n1` | int, the sample size of the first category |
| `n2` | int the sample size of the second category |

### Details

A recursive formula is used:

$$f_{n_1,n_2}(U) = \begin{cases} 0 & \text{if } U < 0 \text{ or } U > n_1 \times n_2 \\ 1 & \text{if } (n_1 = 1 \text{ or } n_2 = 1) \text{ and } 0 \leq U \leq n_1 \times n_2 \\ f_{n_1,n_2-1}(U) + f_{n_1-1,n_2}(U - n_2) & \text{else} \end{cases}$$

This formula is found in Mann and Whitney (1947, p. 51), Dinneen and Blakesley 1973, p. 269) and described also in Festinger (1946).

To convert a W statistic to a U statistic use:

$$U = W - \frac{n_1 \times (n_1 + 1)}{2}$$

### Value

result : a list with the counts starting with the count for U=0

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Dinneen, L. C., & Blakesley, B. C. (1973). Algorithm AS 62: A generator for the sampling distribution of the Mann- Whitney U statistic. *Journal of the Royal Statistical Society. Series C (Applied Statistics), 22*(2), 269–273. doi:10.2307/2346934

Festinger, L. (1946). The significance of difference between means without reference to the frequency distribution function. *Psychometrika, 11*(2), 97–105. doi:10.1007/BF02288926

Mann, H. B., & Whitney, D. R. (1947). On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics, 18*(1), 50–60. doi:10.1214/aoms/1177730491

---

# `di_mwwf` *Mann-Whitney-Wilcoxon Count*

## Description

This function will return the number of possible ways to obtain a specified U value given n1 and n2 cases in each category.

## Usage

```
di_mwwf(u, n1, n2, memo = list())
```

## Arguments

| | |
|---|---|
| `u` | int, the U test statistic |
| `n1` | int, the sample size of the first category |
| `n2` | int the sample size of the second category |
| `memo` | optional list, memoize the result |

## Details

A recursive formula is used:

$$f_{n_1,n_2}(U) = \begin{cases} 0 & \text{if } U < 0 \text{ or } U > n_1 \times n_2 \\ 1 & \text{if } (n_1 = 1 \text{ or } n_2 = 1) \text{ and } 0 \leq U \leq n_1 \times n_2 \\ f_{n_1,n_2-1}(U) + f_{n_1-1,n_2}(U - n_2) & \text{else} \end{cases}$$

This formula is found in Mann and Whitney (1947, p. 51), Dinneen and Blakesley 1973, p. 269) and described also in Festinger (1946).

To convert a W statistic to a U statistic use:

$$U = W - \frac{n_1 \times (n_1 + 1)}{2}$$

## Value

result : a list with the counts starting with the count for U=0

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Dinneen, L. C., & Blakesley, B. C. (1973). Algorithm AS 62: A generator for the sampling distribution of the Mann- Whitney U statistic. *Journal of the Royal Statistical Society. Series C (Applied Statistics), 22*(2), 269–273. doi:10.2307/2346934

Festinger, L. (1946). The significance of difference between means without reference to the frequency distribution function. *Psychometrika, 11*(2), 97–105. doi:10.1007/BF02288926

Mann, H. B., & Whitney, D. R. (1947). On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics, 18*(1), 50–60. doi:10.1214/aoms/1177730491

---

## `di_mwwpmf` *Mann-Whitney-Wilcoxon Probability Mass Function*

### Description

This function returns the probability for the specified U statistic, given n1 and n2 cases in each category.

It first uses the di_mwwf function to determine the count for the u value, and divides it by the total number of possible arrangements.

The dwilcox() function from R's stats library does the same, and is probably more optimized than this function.

### Usage

```
di_mwwpmf(u, n1, n2)
```

### Arguments

| | |
|---|---|
| `u` | int, the U test statistic |
| `n1` | int, the sample size of the first category |
| `n2` | int the sample size of the second category |

### Details

See the details in di_mwwf() on how the frequency is determined. This is then divided by the total number of possibilities, which is the number of ways we can choose $n_1$ items out of $n$, without replacement. This is the binomial coefficient, or number of combinations:

$$C(n, n_1) = nCr(n, n_1) = \binom{n}{n_1} = \frac{n!}{n_1! \times (n - n_1)!}$$

To convert a W statistic to a U statistic use:

$$U = W - \frac{n_1 \times (n_1 + 1)}{2}$$

### Value

result : a list with the counts starting with the count for U=0

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

`di_scdf` *Spearman Rho Distribution*

---

## Description

The Spearman Rank Correlation Coefficient Distribution. Will return a two-tailed p-value

This function makes use of the *pspearman* library for exact computations.

## Usage

```
di_scdf(
  n,
  rs,
  method = c("t", "z-fieller", "z-olds", "iman-conover", "AS89", "exact"),
  iters = 500
)
```

## Arguments

| | |
|---|---|
| `n` | the number of scores (should be equal in both variables) |
| `rs` | the Spearman rank correlation coefficient |
| `method` | the test to be used |
| `iters` | the number of iterations to use, only applicable if Iman-Conover is used |

## Details

The exact distribution is calculated using the following steps:

1. Determine all possible permutations of the scores in the first variable
2. Determine for each permutation the Spearman rho with the second variable
3. Count how often the Spearman rho is above the Spearman rho between the original two variables
4. Divide the results by $n!$

This procedure can be used by using the *he_spearman_exact(ord1, ord2)* function.

This function however, makes use of the *pspearman* function from the *pspearman* library. It seems this uses van de Wiel and Bucchianico (2001) method for the exact distribution, which can handle larger sample sizes (up to n = 22).

The Student t distribution approximation uses (Kendall & Stuart, 1979, p. 503; Iman & Conover, 1978):

$$t_s = r_s \times \sqrt{\frac{n-2}{1-r_s^2}}$$

$$df = n - 2$$

$$sig = 2 \times \left(1 - T\left(\left|t_s\right|, df\right)\right)$$

Iman and Conover refer to Pitman (1937) for the test.

The Fieller's standard normal distribution approximation uses (Fieller et al., 1957, p. 472; Choi, 1977, p. 646):

$$z_F = \sqrt{\frac{n-3}{1.06}} \times \text{atanh}\left(r_s\right)$$

$$sig = 2 \times \left(1 - \Phi\left(\left|z_F\right|\right)\right)$$

The Old's standard normal distribution approximation uses (Olds, 1938, p. 142; Olds, 1949, p. 117):

$$z_O = \frac{x}{ASE}$$

$$sig = 2 \times \left(1 - \Phi\left(\left|z_O\right|\right)\right)$$

With:

$$x = \frac{S}{2} - \frac{n^3 - n}{12}$$

$$ASE = \sqrt{n-1} \times \frac{n^2 + n}{12}$$

$$S = \frac{\left(n^3 - n\right) \times \left(1 - r_s\right)}{6}$$

A combination of the Student t and Normal approximation by Iman and Conover (1978, p. 272) uses:

$$J = \frac{r_s}{2} \times \left(\sqrt{n-1} + \sqrt{n-2}1 - r_s^2\right)$$

And reject the null hypothesis if:

$$J > J_{crit}$$

With:

$$J_{crit} = \frac{Q\left(\Phi\left(1 - \frac{\alpha}{2}\right)\right) + Q\left(T\left(1 - \frac{\alpha}{2}, df\right)\right)}{2}$$

$$df = n - 2$$

The function will use a binary search to find $alpha$ such that $J = J_{crit}$.

One of the more popular methods is Algorithm AS 89 (Best & Roberts, 1975). This is available as a separate helper function *he_AS89(n, S)*.

Often in publications the test statistic $S$ is mentioned, this can be defined as:

$$S = \sum_{i=1}^{n} d_i^2 = \sum_{i=1}^{n} \left(r_{x_i} - r_{y_i}\right)^2$$

Which if there are no ties is equal to:

$$S = \frac{\left(n^3 - n\right) \times \left(1 - r_s\right)}{6}$$

## Value

A dataframe with:

| | |
|---|---|
| `statistic` | the statistic from the test (only if applicable) |
| `df` | the degrees of freedom (only if applicable) |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Best, D. J., & Roberts, D. E. (1975). Algorithm AS 89: The upper tail probabilities of Spearman's rho. *Applied Statistics, 24*(3), 377–379. https://doi.org/10.2307/2347111

Choi, S. C. (1977). Tests of equality of dependent correlation coefficients. *Biometrika, 64*(3), 645–647. https://doi.org/10.1093/biomet/64.3.645

Fieller, E. C., Hartley, H. O., & Pearson, E. S. (1957). Tests for rank correlation coefficients. I. *Biometrika, 44*(3–4), 470–481. https://doi.org/10.1093/biomet/44.3-4.470

Iman, R. L., & Conover, W. J. (1978). Approximations of the critical region for spearman's rho with and without ties present. *Communications in Statistics - Simulation and Computation, 7*(3), 269–282. https://doi.org/10.1080/03610917808812076

Kendall, M., & Stuart, A. (1979). *The advanced theory of statistics. Volume 2: Inference and relationship* (4th ed.). Griffin.

Olds, E. G. (1938). Distributions of sums of squares of rank differences for small numbers of individuals. *The Annals of Mathematical Statistics, 9*(2), 133–148. https://doi.org/10.1214/aoms/1177732332

Olds, E. G. (1949). The 5% significance levels for sums of squares of rank differences and a correction. *The Annals of Mathematical Statistics, 20*(1), 117–118. https://doi.org/10.1214/aoms/1177730099

Pitman, E. J. G. (1937). Significance tests which may be applied to samples from any populations. II. The correlation coefficient test. Supplement to the *Journal of the Royal Statistical Society, 4*(2), 225–232. https://doi.org/10.2307/2983647

van de Wiel, M. A., & Bucchianico, A. D. (2001). Fast computation of the exact null distribution of Spearman's rho and Page's L statistic for samples with and without ties. *Journal of Statistical Planning and Inference, 92*(1–2), 133–145. https://doi.org/10.1016/S0378-3758(00)00166-X

## Examples

```
n = 12
rs = 0.8
di_scdf(n, rs)
di_scdf(n, rs, method="exact")
di_scdf(n, rs, method="t")
di_scdf(n, rs, method="z-fieller")
di_scdf(n, rs, method="z-olds")
di_scdf(n, rs, method="iman-conover")
di_scdf(n, rs, method="AS89")
```

---

`di_wcdf` *Wilcoxon Cumulative Distribution Function*

---

## Description

This function will give the cumulative probability of a sum of ranks of T, given a sample size of n.

Some explanation on this distribution can be found in this YouTube video. This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
di_wcdf(T, n, method = "shift")
```

## Arguments

| | |
|---|---|
| `T` | int with the sum of ranks |
| `n` | int with the sample size |
| `method` | optional the calculation method to use. Either "shift" (default), "enumerate", "recursive". |

## Details

The enumeration method will create all possible combinations of ranks 1 to n, sum each of these, and then determines the count of each unique sum of ranks. It then uses this to determine the probability and cumulative probabilities.

The recursive method uses the formula from McCornack (1965, p. 864):

$$srf\left(x,y\right) = \begin{cases} 0 & x < 0 \\ 0 & x > \binom{y+1}{2} \\ 1 & y = 1 \land \left(x = 0 \lor x = 1\right) \\ srf^*\left(x,y\right) & y \geq 0 \end{cases}$$

with:

$$srf^*\left(x,y\right) = srf\left(x-y,y-1\right) + srf\left(x,y-1\right)$$

The shift-algorithm from Streitberg and Röhmel (1987), and can also be found in Munzel and Brunner (2002). This works as follows.

- Start with listing all values from 0 to the maximum possible sum of ranks, so 0 to (n×(n+1))/2
- Create a vector with the value 1 followed by n times a 0.
- Create a shifted vector by moving all values by 1.
- Add the two results (the original and the shifted version)
- This will be the updated vector
- Shift the vector now by 2
- Add the two results (the updated vector with and the two shifted version)
- Repeat these steps each time shifting by one more than the previous. Stop when n-times shifting has been done.

This Wilcoxon Signed Rank Test (One-Sample) uses this distribution. The `ts_wilcoxon_os` function performs this test, but uses R's own psignrank

## Value

A float with the requested probability

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

McCornack, R. L. (1965). Extended tables of the Wilcoxon matched pair signed rank statistic. *Journal of the American Statistical Association, 60*(311), 864–871. doi:10.2307/2283253

Munzel, U., & Brunner, E. (2002). An exact paired rank test. *Biometrical Journal, 44*(5), 584. doi:10.1002/1521-4036(200207)44:5<584::AID-BIMJ584>3.0.CO;2-9

Streitberg, B., & Röhmel, J. (1987). Exakte Verteilungen für Rang-und Randomisierungstests im allgemeinen c-Stichprobenproblem. *EDV in Medizin und Biologie, 18*(1), 12–19.

## Examples

```
di_wcdf(T=8, n=12)
```

---

# `di_wpmf` *Wilcoxon Probability Mass Function*

## Description

This function will give the probability of a sum of ranks of T, given a sample size of n.

Some explanation on this distribution can be found in this YouTube video. This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
di_wpmf(T, n, method = "shift")
```

## Arguments

| | |
|---|---|
| `T` | int with the sum of ranks |
| `n` | int with the sample size |
| `method` | optional the calculation method to use. Either "shift" (default), "enumerate", "recursive". |

## Details

The enumeration method will create all possible combinations of ranks 1 to n, sum each of these, and then determines the count of each unique sum of ranks. It then uses this to determine the probability.

The recursive method uses the formula from McCornack (1965, p. 864):

$$srf\left(x,y\right)=\begin{cases}0 & x<0\\ 0 & x>\binom{y+1}{2}\\ 1 & y=1\wedge\left(x=0\vee x=1\right)\\ srf^{*}\left(x,y\right) & y\geq 0\end{cases}$$

with:

$$srf^{*}\left(x,y\right)=srf\left(x-y,y-1\right)+srf\left(x,y-1\right)$$

The shift-algorithm from Streitberg and Röhmel (1987), and can also be found in Munzel and Brunner (2002). This works as follows.

- Start with listing all values from 0 to the maximum possible sum of ranks, so 0 to (n×(n+1))/2
- Create a vector with the value 1 followed by n times a 0.
- Create a shifted vector by moving all values by 1.
- Add the two results (the original and the shifted version)
- This will be the updated vector
- Shift the vector now by 2
- Add the two results (the updated vector with and the two shifted version)
- Repeat these steps each time shifting by one more than the previous. Stop when n-times shifting has been done.

## Value

A float with the requested probability

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

@examples di_wpmf(T=8, n=12)

## References

McCornack, R. L. (1965). Extended tables of the Wilcoxon matched pair signed rank statistic. *Journal of the American Statistical Association, 60*(311), 864–871. doi:10.2307/2283253

Munzel, U., & Brunner, E. (2002). An exact paired rank test. *Biometrical Journal, 44*(5), 584. doi:10.1002/1521-4036(200207)44:5<584::AID-BIMJ584>3.0.CO;2-9

Streitberg, B., & Röhmel, J. (1987). Exakte Verteilungen für Rang-und Randomisierungstests im allgemeinen c-Stichprobenproblem. *EDV in Medizin und Biologie, 18*(1), 12–19.

---

`es_alt_ratio` *Alternative Ratio*

---

## Description

The Alternative Ratio is an effect size measure that could be accompanying a one-sample binomial, score or Wald test.It is simply the sample proportion (percentage), divided by the expected population proportion (often set at 0.5)

The Alternative Ratio is only mentioned in the documentation of a program called PASS from NCSS (n.d.), and referred to as Relative Risk by JonB (2015).

This function is shown in this YouTube video and the effect size is also described at PeterStatistics.com

## Usage

```
es_alt_ratio(data, p0 = 0.5, p0Cat = NULL, codes = NULL)
```

## Arguments

| | |
|---|---|
| `data` | vector with the data |
| `p0` | Optional hypothesized proportion for the first category (default is 0.5) |
| `p0Cat` | Optional the category for which p0 was used |
| `codes` | Optional vector with the two codes to use |

## Details

To decide on which category is associated with p0 the following is used:

- If codes are provided, the first code is assumed to be the category for the p0.
- If p0Cat is specified that will be used for p0 and all other categories will be considered as category 2, this means if there are more than two categories the remaining two or more (besides p0Cat) will be merged as one large category.
- If neither codes or p0Cat is specified and more than two categories are in the data a warning is printed and no results.
- If neither codes or p0Cat is specified and there are two categories, p0 is assumed to be for the category closest matching the p0 value (i.e. if p0 is above 0.5 the category with the highest count is assumed to be used for p0)

The formula used is:

$$AR = \frac{p}{\pi}$$

*Symbols used*:

- $p$ is the sample proportion of one of the categories
- $\pi$ the expected proportion

### Value

Dataframe with:

| | |
|---|---|
| `AR1` | the alternative category for one category |
| `AR2` | the alternative category for the other category |
| `comment` | the category for which p0 was |

### Before, After and Alternatives

Before this effect size you might first want to perform a test: `ts_binomial_os`, for One-Sample Binomial Test `ts_score_os`, for One-Sample Score Test `ts_wald_os`, for One-Sample Wald Test

Unfortunately I'm not aware of any rule-of-thumb for this measure.

Alternatives for this effect size could be: `es_cohen_g`, for Cohen g `es_cohen_h_os`, for Cohen h' `r_rosenthal`, for Rosenthal Correlation if a z-value is available

### Author(s)

P. Stikker. Companion Website, YouTube Channel

### References

JonB. (2015, October 14). Effect size of a binomial test and its relation to other measures of effect size. StackExchange - Cross Validated. https://stats.stackexchange.com/q/176856

NCSS. (n.d.). Tests for one proportion. In PASS Sample Size Software (pp. 100-1-100–132). Retrieved November 10, 2018, from https://www.ncss.com/wp-content/themes/ncss/pdf/Procedures/PASS/Tests_for_One_

### Examples

```
# Example 1: Numeric list
ex1 = c(1, 1, 2, 1, 2, 1, 2, 1)
es_alt_ratio(ex1)
es_alt_ratio(ex1, p0=0.3)

# Example 2: Text list
ex2 = c("Female", "Male", "Male", "Female", "Male", "Male")
es_alt_ratio(ex2)
es_alt_ratio(ex2, p0Cat='Female')
es_alt_ratio(ex2, codes=c('Male', 'Female'))

# Example 3: dataframe
file1 <- "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(file1, sep=",", na.strings=c("", "NA"))
es_alt_ratio(df1['sex'])
es_alt_ratio(df1['mar1'], codes=c("DIVORCED", "NEVER MARRIED"))
```

---

## `es_bag_s` *Bennett-Alpert-Goldstein S*

---

### Description

An effect size meaure, that measures the how strongly two raters or variables, agree with each other.

It takes the proportions of cases that both agree, and adjusts for the number of categories. Scott's pi (see es_scott_pi()) does this as well, and improves on this measure.

### Usage

```
es_bag_s(field1, field2, categories = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |

### Details

The formula used (Bennett et al., 1954, p. 307):

$$S = \frac{k}{k-1} \times \left(p_0 - \frac{1}{k}\right)$$

With:

$$P = \sum_{i=1}^{r} F_{i,i}$$

$$p_0 = \frac{P}{n}$$

*Symbols used*

- $F_{i,j}$, the observed count in row i and column j.
- $r$, is the number of rows (categories in the first variable)
- $n$, is the total number of scores

### Value

S, the Bennett-Alpert-Goldstein value

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Bennett, E. M., Alpert, R., & Goldstein, A. C. (1954). Communications through limited response questioning. *Public Opinion Quarterly, 18*(3), 303. doi:10.1086/266520

`es_bin_bin` *Effect Sizes for Binary vs. Binary*

## Description

Various measures of association/agreement/similarity for binary vs. binary cases.

## Usage

```
es_bin_bin(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  method = "odds-ratio"
)
```

## Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with selection and/or order for categories of field1 |
| `categories2` | : optional list with selection and/or order for categories of field2 |
| `method` | : the method to use. Default is odds-ratio |

## Details

The method can be set to any of the following:

| | | | |
|---|---|---|---|
| alroy | ample | anderberg | austin-colwell |
| baroni-urbani-buser-1 | baroni-urbani-buser-2 | becker-clogg-1 | becker-clogg-2 |
| bonett-price-1 | bonett-price-2 | bonett-price-3 | braun-blanquet |
| camp-1 | camp-2 | camp-3 | chen-popovich |
| clement | cohen-kappa | cohen-w | cole-c1 |
| cole-c2 | cole-c3 | cole-c4 | cole-c5 |
| cole-c6 | cole-c7 | cole-c8 | contingency |
| czekanowski | dennis | dice-1 | dice-2 |
| dice-3 | digby | doolittle | driver-kroeber-1 |
| driver-kroeber-2 | edward | eyraud | fager-mcgowan-1 |
| fager-mcgowan-2 | faith | fleiss | forbes-1 |
| forbes-2 | fossum-kaskey | gilbert | gilbert-wells |
| gk-lambda-1 | gk-lambda-2 | gleason | gower |
| gower-legendre | hamann | harris-lahey | hawkins-dotson |
| hurlbert | jaccard | johnson | kent-foster-1 |
| kent-foster-2 | kuder-richardson | kulczynski-1 | kulczynski-2 |
| loevinger | matching | maxwell-pilliner | mcconnaughey |
| mcewen-michael | mountford | nei-li | ochiai-1 |
| ochiai-2 | odds-ratio | otsuka | pearson |
| pearson-heron | pearson-q1 | pearson-q2 | pearson-q3 |

pearson-q4
pearson-q5
peirce-1
peirce-2
peirce-3
phi
rogers-tanimoto
rogot-goldberg
russell-rao
scott
simpson
sokal-michener
sokal-sneath-1
sokal-sneath-2
sokal-sneath-3
sokal-sneath-4
sokal-sneath-5
sorgenfrei
stiles
tanimoto
tarantula
tarwid
tulloss
yule-q
yule-r
yule-y

If we have a 2x2 table with the following values:

| | Column 1 | Column 2 | total |
|---|---|---|---|
| row 1 | $a$ | $b$ | $R_1$ |
| row 2 | $c$ | $d$ | $R_2$ |
| total | $C_1$ | $C_2$ | $n$ |

then the following formula are used:

| Nr | Label | Formula |
|---|---|---|
| 1 | Russell-Rao | $\frac{a}{n}$ |
| 2 | Dice-1 | $\frac{a}{R_1}$ |
| 3 | Dice-2 | $\frac{a}{C_1}$ |
| 4 | Braun-Blanquet | $\frac{a}{\max(R_1,C_1)}$ |
| 5 | Simpson Similarity | $\frac{a}{\min(R_1,C_1)}$ |
| 6 | Kulczynski-1 | $\frac{a}{b+c}$ |
| 7 | Jaccard | $\frac{a}{a+b+c}$ |
| 8 | Sokal-Sneath-1 | $\frac{a}{a+2b+2c}$ |
| 9 | Gleasson | $\frac{2a}{2a+b+c}$ |
| 10 | Mountford | $\frac{2a}{a(b+c)+2bc}$ |
| 11 | Driver-Kroeber | $\frac{a}{\sqrt{R_1 C_1}}$ |
| 12 | Sorgenfrei | $\frac{a^2}{R_1 C_1}$ |
| 13 | Johnson | $\frac{a}{R_1} + \frac{a}{C_1}$ |
| 14 | Kulczynski-2 | $\frac{1}{2}\left(\frac{a}{R_1} + \frac{a}{C_1}\right)$ |
| 15 | Fager-McGowan-1 | $\frac{a}{\sqrt{R_1 C_1}} - \frac{1}{2\sqrt{\max(R_1,C_1)}}$ |
| 16 | Fager-McGowan-2 | $\frac{a}{\sqrt{R_1 C_1}} - \frac{\sqrt{\max(R_1,C_1)}}{2}$ |
| 17 | tarantula | $\frac{aR_2}{cR_1}$ |
| 18 | Ample | $\left\|\frac{aR_2}{cR_1}\right\|$ |
| 19 | Gilbert | $\frac{an-R_1C_1}{C_1n+R_1n-an-R_1C_1}$ |
| 20 | Fossum-Kaskey | $\frac{n\left(a-\frac{1}{2}\right)^2}{R_1C_1}$ |
| 21 | Forbes - 1 | $\frac{na}{R_1C_1}$ |
| 22 | Eyraud | $\frac{a-R_1C_1}{R_1R_2C_1C_2}$ |
| 23 | Sokal-Michener | $\frac{a+d}{n}$ |
| 24 | Faith | $\frac{a+\frac{1}{2}}{n}$ |
| 25 | Sokal-Sneath-5 | $\frac{a+d}{b+c}$ |
| 26 | Rogers-Tanimoto | $\frac{a+d}{a+2(b+c)+d}$ |
| 27 | Sokal-Sneath-2 | $\frac{2a+2d}{2a+b+c+2d}$ |

| | | |
|---|---|---|
| 28 | Gower | $\frac{a+d}{\sqrt{R_1 R_2 C_1 C_2}}$ |
| 29 | Sokal-Sneath-4 | $\frac{ad}{\sqrt{R_1 R_2 C_1 C_2}}$ |
| 30 | Rogot-Goldberg | $\frac{a}{R_1+C_1} + \frac{d}{R_2+C_2}$ |
| 31 | Sokal-Sneath-3 | $\frac{1}{4}\left(\frac{a}{R_1} + \frac{a}{C_1} + \frac{d}{R_2} + \frac{d}{C_2}\right)$ |
| 32 | Hawkin-Dotson | $\frac{1}{2}\left(\frac{a}{a+b+c} + \frac{d}{b+c+d}\right)$ |
| 33 | Clement | $\frac{aR_2}{nR_1} + \frac{dR_1}{nR_2}$ |
| 34 | Harris-Lahey | $\frac{a(R_2+C_2)}{2n(a+b+c)} + \frac{d(R_1+C_1)}{2n(b+c+d)}$ |
| 35 | Austin-Colwell | $\frac{2}{\pi}\text{arcsin}\sqrt{\frac{a+d}{n}}$ |
| 36 | Baroni-Urbani-Buser-1 | $\frac{\sqrt{ad}+a}{\sqrt{ad}+a+b+c}$ |
| 37 | Peirce-1 | $\frac{ad-bc}{R_1 R_2}$ |
| 38 | Peirce-2 | $\frac{ad-bc}{C_1 C_2}$ |
| 39 | Cole C1 | $\frac{ad-bc}{R_1 C_1}$ |
| 40 | Loevinger | $\frac{ad-bc}{\min(R_1 C_2, R_2 C_1)}$ |
| 41 | Cole C7 | $\begin{cases} \frac{ad-bc}{R_1 C_2} & \text{if } ad \geq bc \\ \frac{ad-bc}{R_1 C_1} & \text{if } ad < bc \text{ and } a \leq d \\ \frac{ad-bc}{R_2 C_2} & \text{if } ad < bc \text{ and } a > d \end{cases}$ |
| 42 | Dennis | $\frac{ad-bc}{\sqrt{nR_1 C_1}}$ |
| 43 | Phi | $\frac{ad-bc}{\sqrt{R_1 R_2 C_1 C_2}}$ |
| 44 | Doolittle | $\frac{(ad-bc)^2}{R_1 R_2 C_1 C_2}$ |
| 45 | Peirce-3 | $\frac{ad-bc}{ab+2bc+cd}$ |
| 46 | Cohen-kappa | $\frac{2(ad-bc)}{R_1 C_2 + R_2 C_1}$ |
| 47 | McEwen-Michael | $\frac{4(ad-bc)}{(a+d)^2+(b+c)^2}$ |
| 48 | Kuder-Richardson | $\frac{4(ad-bc)}{R_1 R_2 + C_1 C_2 + 2ad - 2bc}$ |
| 49 | Scott | $\frac{4ad-(b+c)^2}{(R_1+C_1)(R_2+C_2)}$ |
| 50 | Maxwell-Pilliner | $\frac{2(ad-bc)}{R_1 R_2 + C_1 C_2}$ |
| 51 | Cole C5 | $\frac{\sqrt{2}(ad-bc)}{\sqrt{(ad-bc)^2 + R_1 R_2 C_1 C_2}}$ |
| 52 | Hamann | $\frac{(a+d)-(b+c)}{n}$ |
| 53 | Fleiss | $\frac{(ad-bc)(R_1 C_2 + R_2 C_1)}{2R_1 R_2 C_1 C_2}$ |
| 54 | Yule Q | $\frac{ad-bc}{ad+bc}$ |
| 55 | Yule Y | $\frac{\sqrt{ad}-\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}$ |
| 56 | Digby H | $\frac{(ad)^{3/4}-(bc)^{3/4}}{(ad)^{3/4}+(bc)^{3/4}}$ |
| 57 | Edward Q | $\frac{OR^{\pi/4}-1}{OR^{\pi/4}+1}$ |
| 58 | Tarwid | $\frac{na-R_1 C_1}{na+R_1 C_1}$ |
| 59 | Bonett-Price-1 | $\frac{\hat{w}^x-1}{\hat{w}^x+1}$ |
| 60 | Contingency coefficient | $\sqrt{\frac{\chi^2}{n+\chi^2}}$ |
| 61 | Cohen w | $\sqrt{\frac{\chi^2}{\chi^2}}$ |
| 62 | Pearson | $\sqrt{\frac{\phi^2}{n+\phi^2}}$ |
| 63 | Hurlbert | $\frac{ad-bc}{\|ad-bc\|}\sqrt{\frac{\chi^2-\chi^2_{min}}{\chi^2_{max}-\chi^2_{min}}}$ |
| 64 | Stiles | $\log_{10}\left(\frac{n\left(\|ad-bc\|-\frac{n}{2}\right)^2}{R_1 R_2 C_1 C_2}\right)$ |
| 65 | McConnaughey | $\frac{a^2-bc}{R_1 C_1}$ |

| | | |
|---|---|---|
| 66 | Baroni-Urbani-Buser-2 | $\frac{a-b-c+\sqrt{ad}}{a+b+c+\sqrt{ad}}$ |
| 67 | Kent-Foster-1 | $\frac{-bc}{bR_1+cC_1+bc}$ |
| 68 | Kent-Foster-2 | $\frac{-bc}{bR_2+cC_2+bc}$ |
| 69 | Tulloss | $\sqrt{U \times S \times R}$ |
| 70 | Gilbert-Wells | $\ln\left(a\right) - \ln\left(n\right) - \ln\left(\frac{R_1}{n}\right) - \ln\left(\frac{C_1}{n}\right)$ |
| 71 | Yule r | $\cos\left(\frac{\pi\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}\right)$ |
| 72 | Anderberg | $\frac{\sigma-\sigma'}{2n}$ |
| 73 | Alroy F | $\frac{a\left(n'+\sqrt{n'}\right)}{a\left(n'+\sqrt{n'}\right)+\frac{3}{2}bc}$ |
| 74 | Pearson Q1 | $\sin\left(\frac{\pi}{2} \times \frac{R_2C_1}{R_1C_2}\right)$ |
| 75 | Goodman-Kruskal Lambda-1 | $\frac{\sigma-\sigma'}{2n-\sigma'}$ |
| 76 | Goodman-Kruskal Lambda-2 | $\frac{2\min(a,d)-b-c}{2\min(a,d)+b+c}$ |
| 77 | Odds Ratio | $\frac{ad}{bc}$ |
| 78 | Pearson Q4 | $\sin\frac{\pi}{2} \times \frac{1}{1+\frac{2bcn}{(ad-bc)(b+c)}}$ |
| 79 | Pearson Q5 | $\sin\frac{\pi}{2} \times \frac{1}{\sqrt{1+\frac{4abcdn^2}{(ad-bc)^2(a+d)(b+c)}}}$ |
| 80 | Camp (3 ver.) | $\frac{m}{\sqrt{1+\Theta \times m^2}}$ |
| 81 | Becker-Clogg-1 | $\frac{g-1}{g+1}$ |
| 82 | Becker-Clogg-2 | $\frac{OR^{13.3/\delta}-1}{OR^{13.3/\delta}+1}$ |
| 83 | Bonett-Price-r | $\cos\left(\frac{\pi}{1+\omega^c}\right)$ |
| 84 | Bonett-Price-rhat | $\cos\left(\frac{\pi}{1+\hat{\omega}^{\hat{c}}}\right)$ |
| 85 | Chen-Popovich | $\frac{ad-bc}{\lambda_x\lambda_y n^2}$ |

**Equation 57**

$$OR = \frac{ad}{bc}$$

**Equation 59**

$$x = \frac{1}{2} - \left(\frac{1}{2} - p_{min}\right)^2$$

$$p_{min} = \frac{\min\left(R_1, R_2, C_1, C_2\right)}{n}$$

$$\hat{\omega} = \frac{(a + 0.1) \times (d + 0.1)}{(b + 0.1) \times (c + 0.1)}$$

**Equations 60, 61, and 63**

$$\chi^2 = \frac{n\left(ad - bc\right)^2}{R_1R_2C_1C_2}$$

**Equation 62**

$$\Phi = \frac{|ad - bc|}{\sqrt{R_1R_2C_1C_2}}$$

Note that Choi et. al ommit the absolute value, but this would create problems with taking the square root if bc>ad.

**Equation 63:**

$$\chi_{max}^2 = \begin{cases} \frac{nR_1C_2}{R_2C_1} & \text{if } ad \geq bc \\ \frac{nR_1C_1}{R_2C_2} & \text{if } ad < bc \text{ and } a \leq d \\ \frac{nR_2C_2}{R_1C_1} & \text{if } ad < bc \text{ and } a > d \end{cases}$$

$$\chi_{min}^2 = \frac{n^3\left(\hat{a} - g\left(\hat{a}\right)\right)^2}{R_1R_2C_1C_2}$$

$$\hat{a} = \frac{R_1C_1}{n}$$

$$g\left(\hat{a}\right) = \begin{cases} \lfloor\hat{a}\rfloor & \text{if } ad < bc \\ \lceil\hat{a}\rceil & \text{if } ad \geq bc \end{cases}$$

**Equation 69**

$$U = \log_2\left(1 + \frac{\min\left(b,c\right) + a}{\max\left(b,c\right) + a}\right)$$

$$S = \frac{1}{\sqrt{\log_2\left(2 + \frac{\min(b,c)}{a+1}\right)}}$$

$$R = \log_2\left(1 + \frac{a}{R_1}\right)\log_2\left(1 + \frac{a}{RC1}\right)$$

**Equation 71 and 75**

$$\sigma = \max\left(a,b\right) + \max\left(c,d\right) + \max\left(a,c\right) + \max\left(b,d\right)$$

$$\sigma' = \max\left(R_1,R_2\right) + \max\left(C_1,C_2\right)$$

**Equation 73**

$$n' = a + b + c$$

**Equation 80**

Camp (1934, pp. 309) describes the following steps for the calculation: Step 1: If total of column 1 (C1) is less than column 2 (C2), swop the two columns

Step 2: Calculate $p = \frac{C1}{n}$, $p_1 = \frac{a}{n}$, and $p_2 = \frac{c}{C2}$

Step 3: Determine $z_1$, $z_2$ as the normal deviate corresponding to the area $p_1$, $p_2$ resp. (inverse standard normal cumulative distribution)

Step 4: Determine y the normal ordinate corresponding to $p$ (the height of the normal distribution)

Step 5: Calculate $m = \frac{p\times(1-p)\times(z_1+z_2)}{y}$

Step 6: Find phi in a table of phi values

Camp suggested for a very basic approximation to simply use $\phi = 1$.

For a better approximation Camp made the following table:

| p | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|---|---|---|---|---|---|
| phi | 0.637 | 0.63 | 0.62 | 0.60 | 0.56 |

Cureton (1968, p. 241) expanded on this table and produced:

| p | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.5 | 0.637 | 0.636 | 0.636 | 0.635 | 0.635 | 0.634 | 0.634 | 0.633 | 0.633 | 0.632 | 0.631 |
| 0.6 | 0.631 | 0.631 | 0.630 | 0.629 | 0.628 | 0.627 | 0.626 | 0.625 | 0.624 | 0.622 | 0.621 |
| 0.7 | 0.621 | 0.620 | 0.618 | 0.616 | 0.614 | 0.612 | 0.610 | 0.608 | 0.606 | 0.603 | 0.600 |
| 0.8 | 0.600 | 0.597 | 0.594 | 0.591 | 0.587 | 0.583 | 0.579 | 0.574 | 0.569 | 0.564 | 0.559 |

Step 7: Calculate $r_t = \frac{m}{\sqrt{1+\phi \times m^2}}$

Cureton (1968) describes quite a few shortcomings with this approximation, and circumstances when it might be appropriate.

**Equation 81 and 82**

Version 81 will calculate:

$$\rho^* = \frac{g-1}{g+1}$$

Version 82 will calculate:

$$\rho^{**} = \frac{OR^{13.3/\Delta} - 1}{OR^{13.3/\Delta} + 1}$$

With:

$$g = e^{12.4\times\phi - 24.6\times\phi^3}$$

$$\phi = \frac{\ln\left(OR\right)}{\Delta}$$

$$OR = \frac{\left(\frac{a}{c}\right)}{\left(\frac{b}{d}\right)} = \frac{a\times d}{b\times c}$$

$$\Delta = \left(\mu_{R1} - \mu_{R2}\right) \times \left(v_{C1} - v_{C2}\right)$$

$$\mu_{R1} = \frac{-e^{-\frac{t_r^2}{2}}}{p_{R1}}, \mu_{R2} = \frac{e^{-\frac{t_r^2}{2}}}{p_{R2}}$$

$$v_{C1} = \frac{-e^{-\frac{t_c^2}{2}}}{p_{C1}}, v_{C2} = \frac{e^{-\frac{t_c^2}{2}}}{p_{C2}}$$

$$t_r = \Phi^{-1}\left(p_{R1}\right), t_c = \Phi^{-1}\left(p_{C1}\right)$$

$$p_x = \frac{x}{n}$$

**Equations 83 and 84**

Formula for version 1 is (Bonett & Price, 2005, p. 216):

$$\rho^* = \cos\left(\frac{\pi}{1+\omega^c}\right)$$

With:

$$\omega = OR = \frac{a\times d}{b\times c}$$

$$c = \frac{1 - \frac{|R_1 - C_1|}{5 \times n} - \left(\frac{1}{2} - p_{min}\right)^2}{2}$$

$$p_{min} = \frac{\text{MIN}\left(R_1, R_2, C_1, C_2\right)}{n}$$

Formula for version 2 is (Bonett & Price, 2005, p. 216):

$$\hat{\rho}^* = \cos\left(\frac{\pi}{1 + \hat{\omega}^{\hat{c}}}\right)$$

with:

$$\hat{\omega} = \frac{\left(a + \frac{1}{2}\right) \times \left(d + \frac{1}{2}\right)}{\left(b + \frac{1}{2}\right) \times \left(c + \frac{1}{2}\right)}$$

$$\hat{c} = \frac{1 - \frac{|R_1 - C_1|}{5 \times (n+2)} - \left(\frac{1}{2} - \hat{p}_{min}\right)^2}{2}$$

$$\hat{p}_{min} = \frac{\text{MIN}\left(R_1, R_2, C_1, C_2\right) + 1}{n + 2}$$

**Equation 85**

$$\lambda_x = \Phi^{-1}\left(\frac{R_1}{n}\right)$$

$$\lambda_y = \Phi^{-1}\left(\frac{C_1}{n}\right)$$

with $\Phi^{-1}\left(\ldots\right)$ being the inverse standard normal cumulative distribution

**Sources for formulas**

The formulas were obtained from the following sources. The columns W-C-H show which equation corresponds to my label in:

- W: Warrens (2008, pp. 219–222). Equation 4 from Warrens is the chi-square value and not added.
- C: Choi et al. (2010, pp. 44–45). Equations not added from this source are: Eq. 4 is a '3w Jaccard', could not find a source for this and not added. Equation 12 is just the intersection (a), eq. 13 the innerproduct (a+d), and 66 the dispersion. Equation 51 is the chi-square value and measures 15 to 30 and 62 are just distance measures.
- H: Hubálek (Hubálek, 1982, pp. 671–673)

If no page number is listed in the original source, the formula was taken from Warrens, Choi et al. and/or Hubálek.

| nr | Label | Original | W | C | H |
|---|---|---|---|---|---|
| 1 | Russell-Rao | (Russell & Rao, 1940) | 15 | 14 | 14 |
| 2 | Dice-1 | (Dice, 1945, p. 302) | 17a | | |
| 3 | Dice-2 | (Dice, 1945, p. 302) | 17b | | |
| 4 | Braun-Blanquet | (Braun-Blanquet, 1932) | 12 | 46 | 1 |
| 5 | Simpson Similarity | (Simpson, 1943, p. 20, 1960, p. 301) | 16 | 45 | 2 |
| 6 | Kulczynski-1 | (Kulczynski, 1927) | 11b | 64 | 3 |
| 7 | Jaccard = | (Jaccard, 1901, 1912, p. 39) | 6 | 1 | 4 |
| | Tanimoto | (Tanimoto, 1958, p. 5) | | 65 | |
| 8 | Sokal-Sneath-1 | (Sokal & Sneath, 1963, p. 129) | 30a | 6 | 6 |

| | | | | | |
|---|---|---|---|---|---|
| 9 | Gleasson = | (Gleason, 1920, p. 31) | 9 | | 5 |
| | Dice-3 = | (Dice, 1945, p. 302) | | 2 | |
| | Nei-Li = | (Nei & Li, 1979, p. 5270) | | 5 | |
| | Czekanowski | | | 3 | |
| 10 | Mountford | (Mountford, 1962, p. 45) | 28 | 37 | 15 |
| 11 | Driver-Kroeber = | (Driver & Kroeber, 1932, p. 219) | 13 | 31 | 11 |
| | Ochiai-1 = | (Ochiai, 1957) | | 33 | |
| | Otsuka | (Otsuka, 1936) | | 38 | |
| 12 | Sorgenfrei | (Sorgenfrei, 1958) | 23 | 36 | 12 |
| 13 | Johnson | (Johnson, 1967) | 33 | 43 | 9 |
| 14 | Driver-Kroeber-2 = | (Driver & Kroeber, 1932, p. 219) | 11a | 42 | 8 |
| | Kulczynski-2 | (Kulczynski, 1927) | | 41 | 7 |
| 15 | Fager-McGowan-1 | (Fager & McGowan, 1963, p. 454) | 29 | | |
| 16 | Fager-McGowan-2 | (Fager & McGowan, 1963, p. 454) | | 47 | 13 |
| 17 | tarantula | (Jones & Harrold, 2005) | | 75 | |
| 18 | ample | | | 76 | |
| 19 | Gilbert | (Gilbert, 1884, p. 171) | | | |
| 20 | Fossum-Kaskey | (Fossum & Kaskey, 1966, p. 65) | | 35 | |
| 21 | Forbes - 1 | (Forbes, 1907, p. 279) | 5 | 34 | 40 |
| 22 | Eyraud | (Eyraud, 1936) | | 74 | 17 |
| 23 | Sokal-Michener | (Sokal & Michener, 1958, p. 1417) | 22 | 7 | 20 |
| 24 | Faith | (Faith, 1983, p. 290) | | 10 | |
| 25 | Sokal-Sneath-5 | (Sokal & Sneath, 1963, p. 129) | 30e | 56 | 19 |
| 26 | Rogers-Tanimoto | (Rogers & Tanimoto, 1960) | 25 | 9 | 23 |
| 27 | Sokal-Sneath-2 = | (Sokal & Sneath, 1963, p. 129) | 30b | 8 | 22 |
| | Gower-Legendre | (Gower & Legendre, 1986) | | 11 | |
| 28 | Gower | (Gower, 1971) | | 50 | |
| 29 | Sokal-Sneath-4 = | (Sokal & Sneath, 1963, p. 130) | 30d | 57 | 25 |
| | Ochiai-2 | (Ochiai, 1957) | | 60 | |
| 30 | Rogot-Goldberg | (Rogot & Goldberg, 1966, p. 997) | 32 | | |
| 31 | Sokal-Sneath-3 | (Sokal & Sneath, 1963, p. 130) | 30c | 49 | 18 |
| 32 | Hawkin-Dotson | (Hawkins & Dotson, 1975, pp. 372–373) | 34 | | |
| 33 | Clement | (Clement, 1976, p. 258) | 37 | | |
| 34 | Harris-Lahey | (Harris & Lahey, 1978, p. 526) | 40 | | |
| 35 | Austin-Colwell | (Austin & Colwell, 1977, p. 205) | | | 21 |
| 36 | Baroni-Urbani-Buser-1 | (Baroni-Urbani & Buser, 1976, p. 258) | 38a | 71 | 32 |
| 37 | Peirce-1 | (Peirce, 1884, p. 453) | 1a | | |
| 38 | Peirce-2 | (Peirce, 1884, p. 453) | 1b | | 26 |
| 39 | Cole C1 | (Cole, 1949, p. 415) | | | |
| 40 | Loevinger = | (Loevinger, 1947, p. 30) | 18 | | |
| | Forbes 2 | (Forbes, 1925) | | 48 | 42 |
| 41 | Cole C7 | (Cole, 1949, p. 420) | 19 | | 34 |
| 42 | Dennis | (Dennis, 1965, p. 69) | | 44 | |
| 43 | Pearson Phi = | (Pearson, 1900a, p. 12) | 7 | 54 | 30 |
| | Yule Phi = | (Yule, 1912, p. 596) | | | |
| | Cole C2 | (Cole, 1949, p. 415) | | | |
| 44 | Doolittle | (Doolittle, 1885, p. 123) | 2 | | 31 |
| 45 | Peirce-3 | (Peirce, 1884) | | 73 | 16 |
| 46 | Cohen-kappa | (Cohen, 1960, p. 40) | 24 | | |
| 47 | McEwen-Michael = | (Michael, 1920, p. 57) | 10 | 68 | 39 |
| | Cole C3 | (Cole, 1949, p. 415) | | | |
| 48 | Kuder-Richardson | (Kuder & Richardson, 1937) | 14 | | |

| | | | | | |
|---|---|---|---|---|---|
| 49 | Scott | (Scott, 1955, p. 324) | 21 | | |
| 50 | Maxwell-Pilliner | (Maxwell & Pilliner, 1968) | 35 | | |
| 51 | Cole C5 | (Cole, 1949, p. 416) | | 58 | 29 |
| 52 | Hamann | (Hamann, 1961) | 27 | 67 | 24 |
| 53 | Fleiss | (Fleiss, 1975, p. 656) | 36 | | |
| 54 | Yule Q = | (Yule, 1900, p. 272) | 3 | 61 | 36 |
| | Cole C4 = | (Cole, 1949, p. 415) | | | |
| | Pearson Q2 | (Pearson, 1900, p. 15) | | | |
| 55 | Yule Y | (Yule, 1912, p. 592) | 8 | 63 | 37 |
| 56 | Digby H | (Digby, 1983, p. 754) | 41 | | |
| 57 | Edward Q | (Edwards, 1957; Becker & Clogg, 1988, p. 409) | | | |
| 58 | Tarwid | (Tarwid, 1960, p. 117) | | 40 | 43 |
| 59 | Bonett-Price-1 | (Bonett & Price, 2007, p. 433) | | | |
| 60 | Contingency | (Pearson, 1904, p. 9) | | 52 | 28 |
| 61 | Cohen w | (Cohen, 1988, p. 216) | | | |
| 62 | Pearson | (Pearson, 1904) | | 53 | |
| 63 | Hurlbert / Cole C8 | (Hurlbert, 1969, p. 1) | | | 35 |
| 64 | Stiles | (Stiles, 1961, p. 272) | 26 | 59 | |
| 65 | McConnaughey | (McConnaughey, 1964) | 31 | 39 | 10 |
| 66 | Baroni-Urbani-Buser-2 | (Baroni-Urbani & Buser, 1976, p. 258) | 38b | 72 | 33 |
| 67 | Kent-Foster-1 | (Kent & Foster, 1977, p. 311) | 39a | | |
| 68 | Kent-Foster-2 | (Kent & Foster, 1977, p. 311) | 39b | | |
| 69 | Tulloss | (Tulloss, 1997, p. 133) | | | |
| 70 | Gilbert-Wells | (Gilbert & Wells, 1966) | | | |
| 71 | Yule r | (Yule, 1900, p. 276) | | | |
| | Pearson-Q3 | (Pearson, 1900a, p. 16) | | | |
| | Cole C6 | (Cole, 1949, p. 416) | | | |
| | Pearson-Heron | (Pearson & Heron, 1913) | | 55 | 38 |
| 72 | Anderberg | (Anderberg, 1973) | | 70 | |
| 73 | Alroy F | (Alroy, 2015, eq. 6) | | | |
| 74 | Pearson Q1 | (Pearson, 1900a, p. 15) | | | |
| 75 | Goodman-Kruskal Lambda-1 | (Goodman & Kruskal, 1954, p. 743) | | 69 | |
| 76 | Goodman-Kruskal Lambda-2 | (Goodman & Kruskal, 1954) | 20 | | |
| 77 | Odds Ratio | (Fisher, 1935, p. 50) | | | |
| 78 | Pearson Q4 | (Pearson, 1900, p. 16) | | | |
| 79 | Pearson Q5 | (Pearson, 1900, p. 16) | | | |
| 80 | Camp | (Camp, 1934, p. 309) | | | |
| 81 | Becker-Clogg-1 | (Becker & Clogg, 1988, pp. 410–412) | | | |
| 82 | Becker-Clogg-2 | (Becker & Clogg, 1988, pp. 410–412) | | | |
| 83 | Bonett-Price-2 | (Bonett & Price, 2005, p. 216) | | | |
| 84 | Bonett-Price-3 | (Bonett & Price, 2005, p. 216) | | | |
| 85 | Ched-Popovich | (Chen & Popovich, 2002, p. 37) | | | |

## Value

the effect size measure value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

**References**

Alroy, J. (2015). A new twist on a very old binary similarity coefficient. *Ecology, 96*(2), 575–586. doi:10.1890/14-0471.1

Anderberg, M. R. (1973). *Cluster analysis for applications*. New York, NY, Academic Press.

Austin, B., & Colwell, R. R. (1977). Evaluation of some coefficients for use in numerical taxonomy of microorganisms. *International Journal of Systematic Bacteriology, 27*(3), 204–210. doi:10.1099/00207713-27-3-204

Baroni-Urbani, C., & Buser, M. W. (1976). Similarity of binary data. *Systematic Zoology, 25*(3), 251–259. doi:10.2307/2412493

Becker, M. P., & Clogg, C. C. (1988). A note on approximating correlations from Odds Ratios. *Sociological Methods & Research, 16*(3), 407–424. doi:10.1177/0049124188016003003

Bonett, D. G., & Price, R. M. (2005). Inferential methods for the tetrachoric correlation coefficient. *Journal of Educational and Behavioral Statistics, 30*(2), 213–225. https://doi.org/10.3102/10769986030002213

Bonett, D. G., & Price, R. M. (2007). Statistical inference for generalized yule coefficients in 2 x 2 contingency tables. *Sociological Methods & Research, 35*(3), 429–446. doi:10.1177/0049124106292358

Braun-Blanquet, J. (1932). *Plant sociology: The study of plant communities*. McGraw Hill.

Camp, B. H. (1934). *Mathematical part of elementary statistics*. D.C. Heath and Company, London.

Chen, P. Y., & Popovich, P. M. (2002). *Correlation: Parametric and nonparametric measures*. Sage Publications.

Choi, S.-S., Cha, S.-H., & Tappert, C. (2010). A survey of binary similarity and distance measures. *Journal on Systemics, Cybernetics and Informatics, 8*(1), 43–48.

Clement, P. W. (1976). A formula for computing inter-observer agreement. *Psychological Reports, 39*(1), 257–258. doi:10.2466/pr0.1976.39.1.257

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46. doi:10.1177/001316446002000104

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Cole, L. C. (1949). *The measurement of interspecific associaton. Ecology, 30*(4), 411–424. doi:10.2307/1932444

Cureton, E. E. (1968). Tetrachoric correlation by the Camp approximation. *Educational and Psychological Measurement, 28*(2), 239–244. doi:10.1177/001316446802800202

Dennis, S. F. (1965). The construction of a thesaurus automatically from a sample of text. In M. E. Stevens, V. E. Giuliano, & L. B. Heilprin (Eds.), *Statistical Association Methods for Mechanized Documentation* (Vol. 14, pp. 61–148). U.S. Government Printing Office.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology, 26*(3), 297–302. doi:10.2307/1932409

Digby, P. G. N. (1983). Approximating the tetrachoric correlation coefficient. *Biometrics, 39*(3), 753–757. doi:10.2307/2531104

Doolittle, M. H. (1885). The verification of predictions. *Bulletin of the Philosophical Society of Washington, 7*, 122–127.

Driver, H. E., & Kroeber, A. L. (1932). Quantitative expression of cultural relationships. *University Of California Publications in American Archeology and Ethnology, 31*(4), 211–256.

Edwards, J. H. (1957). A note on the practical interpretation of 2 x 2 tables. *Journal of Epidemiology & Community Health, 11*(2), 73–78. doi:10.1136/jech.11.2.73

Edwards, J. H., & Edwards, A. W. F. (1984). Approximating the tetrachoric correlation coefficient. *Biometrics, 40*(2), 563–563.

Eyraud, H. (1936). Les principes de la mesure des correlations. *Ann. Univ. Lyon, III. Ser., Sect. A, 1*(30–47), 111.

Fager, E. W., & McGowan, J. A. (1963). Zooplankton species groups in the north pacific: Co-occurrences of species can be used to derive groups whose members react similarly to water-mass types. *Science, 140*(3566), 453–460. doi:10.1126/science.140.3566.453

Faith, D. P. (1983). Asymmetric binary similarity measures. *Oecologia, 57*(3), 287–290. doi:10.1007/BF00377169

Fisher, R. A. (1935). The logic of inductive inference. *Journal of the Royal Statistical Society, 98*(1), 39–82. doi:10.2307/2342435

Fleiss, J. L. (1975). Measuring agreement between two judges on the presence or absence of a trait. *Biometrics, 31*(3), 651–659. doi:10.2307/2529549

Forbes, S. A. (1907). On the local distribution of certain Illinois fishes: An essay in statistical ecology. *Illinois Natural History Survey Bulletin, 7*(8), 273–303.

Forbes, S. A. (1925). Method of determining and measuring the associative relations of species. *Science, 61*, 524.

Fossum, E. G., & Kaskey, G. (1966). Optimization and standardization of information retrieval language and systems (AF49(638)1194; p. 96). Univac Division.

Gilbert, G. K. (1884). Finley's tornado predictions. *American Meteorological Journal, 1*(5), 166–172.

Gilbert, N., & Wells, T. C. E. (1966). *Analysis of quadrat data. Journal of Ecology, 54*(3), 675–685. doi:10.2307/2257810

Gleason, H. A. (1920). Some applications of the quadrat method. *Bulletin of the Torrey Botanical Club, 47*(1), 21–33. doi:10.2307/2480223

Glen, S. (2017, August 16). Gamma Coefficient (Goodman and Kruskal's Gamma) & Yule's Q. Statistics How To. https://www.statisticshowto.com/gamma-coefficient-goodman-kruskal/

Goodman, L. A., & Kruskal, W. H. (1954). Measures of association for cross classifications. *Journal of the American Statistical Association, 49*(268), 732–764. doi:10.2307/2281536

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics, 27*(4), 857. doi:10.2307/2528823

Gower, J. C., & Legendre, P. (1986). Metric and Euclidean properties of dissimilarity coefficients. *Journal of Classification, 3*(1), 5–48. doi:10.1007/BF01896809

Hamann, U. (1961). Merkmalsbestand und verwandtschaftsbeziehungen der farinosae: Ein beitrag zum system der monokotyledonen. *Willdenowia, 2*(5), 639–768.

Harris, F. C., & Lahey, B. B. (1978). A method for combining occurrence and nonoccurrence inter-observer agreement scores. *Journal of Applied Behavior Analysis, 11*(4), 523–527. doi:10.1901/jaba.1978.11-523

Hawkins, R. P., & Dotson, V. A. (1975). Reliability scores that delude: An Alice in wonderland trip through the misleading characteristics of inter-observer agreement scores in interval recording. In E. Ramp & G. Semb (Eds.), *Behavior analysis: Areas of research and application *(pp. 359–376). Prentice Hall.

Hubálek, Z. (1982). Coefficients of association and similarity, based on binary (presence-absence) data: An evaluation. *Biological Reviews, 57*(4), 669–689. doi:10.1111/j.1469-185X.1982.tb00376.x

Hurlbert, S. H. (1969). A coefficient of interspecific assciation. *Ecology, 50*(1), 1–9. doi:10.2307/1934657

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin Del La Société Vaudoise Des Sciences Naturelles, 37*, 547–579.

Jaccard, P. (1912). The distribution of the flora in the alpine zone. *The New Phytologist, 11*(2), 37–50.

Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika, 32*(3), 241–254. doi:10.1007/BF02289588

Jones, J. A., & Harrold, M. J. (2005). Empirical evaluation of the tarantula automatic fault-localization technique. Proceedings of the 20th IEEE/ACM International Conference on Automated Software Engineering, 273–282. doi:10.1145/1101908.1101949

Kent, R. N., & Foster, S. L. (1977). Direct observational procedures: Methodological issues in naturalistic settings. In A. R. Ciminero, K. S. Calhoun, & H. E. Adams (Eds.), *Handbook of behavioral assessment* (pp. 279–328). New York, NY: Wiley. http://archive.org/details/handbookofbehavi00cimi

Kuder, G. F., & Richardson, M. W. (1937). The theory of the estimation of test reliability. *Psychometrika, 2*(3), 151–160. doi:10.1007/BF02288391

Kulczynski, S. (1927). Die Pflanzenassoziationen der Pieninen. *Bulletin International de l'Academie Polonaise Des Sciences et Des Lettres, Classe Des Sciences Mathematiques et Naturelles, B (Sciences Naturelles)*, II, 57–203.

Ling, M. H. T. (2010). COPADS, I: Distance coefficients between two lists or sets. *The Python Papers Source Codes, 2*(2), 1–31.

Loevinger, J. (1947). A systematic approach to the construction and evaluation of tests of ability. *Psychological Monographs, 61*(4), i–49. doi:10.1037/h0093565

Maxwell, A. E., & Pilliner, A. E. (1968). Deriving coefficients of reliability and agreement for ratings. *The British Journal of Mathematical and Statistical Psychology, 21*(1), 105–116. doi:10.1111/j.2044-8317.1968.tb00401.x

McConnaughey, B. H. (1964). The determination and analysis of plankton communities. *Marine Research, 7*, 1–40.

Michael, E. L. (1920). Marine Ecology and the coefficient of association: A plea in behalf of quantitative biology. *Journal of Ecology, 8*(1), 54–59. doi:10.2307/2255213

Mountford, M. D. (1962). An index of similarity and its application to classification problems. In P. W. Murphy & D. Phil (Eds.), *Progress in soil zoology* (pp. 43–50). Butterworths.

Nei, M., & Li, W. H. (1979). Mathematical model for studying genetic variation in terms of restriction endonucleases. *Proceedings of the National Academy of Sciences, 76*(10), 5269–5273. doi:10.1073/pnas.76.10.5269

Ochiai, A. (1957). Zoogeographical studies on the soleoid fishes found in Japan and its neighbouring regions-I. *Nippon Suisan Gakkaishi, 22*(9), 522–525. doi:10.2331/suisan.22.522

Otsuka, Y. (1936). The faunal character of the Japanese Pleistocene marine Mollusca, as evidence of the climate having become colder during the Pleistocene in Japan. *Bulletin of the Biogeographical Society of Japan, 6*(16), 165–170.

Pearson, K. (1900). Contributions to the mathematical theory of evolution. VII. On the correlation of characters not quantitatively measurable. *Philosophical Transactions of the Royal Society of London, 195*, 1–405. doi:10.1098/rsta.1900.0022

Pearson, K. (1904). *Contributions to the Mathematical Theory of Evolution. XIII. On the theory of contingency and its relation to association and normal correlation*. Dulau and Co.

Pearson, K., & Heron, D. (1913). On theories of association. *Biometrika, 9*(1/2), 159–315. doi:10.2307/2331805

Peirce, C. S. (1884). The numerical measure of the success of predictions. *Science, 4*(93), 453–454. doi:10.1126/science.ns-4.93.453-a

Rogers, D. J., & Tanimoto, T. T. (1960). A computer program for classifying plants. *Science, 132*(3434), 1115–1118. doi:10.1126/science.132.3434.1115

Rogot, E., & Goldberg, I. D. (1966). A proposed index for measuring agreement in test-retest studies. *Journal of Chronic Diseases, 19*(9), 991–1006. doi:10.1016/0021-9681(66)90032-4

Russell, P. F., & Rao, T. R. (1940). On habitat and association of species of anopheline larvae in south-eastern Madras. *Journal of the Malaria Institute of India, 3*(1), 153–178.

Scott, W. A. (1955). Reliability of content analysis: The case of nominal scale coding. *The Public Opinion Quarterly, 19*(3), 321–325.

Simpson, G. G. (1943). Mammals and the nature of continents. *American Journal of Science, 241*(1), 1–31. doi:10.2475/ajs.241.1.1

Simpson, G. G. (1960). Notes on the measurement of faunal resemblance. *American Journal of Science, 258-A*, 300–311.

Sokal, P. H. A., & Sneath, R. R. (1963). Principles of numerical taxonomy. W.H. Freeman and Company.

Sokal, R., & Michener, C. (1958). A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin.

Sorgenfrei, T. (1958). Molluscan assemblages from the marine middle miocene of south Jutland and their Environments. Vol. I. *Danmarks Geologiske Undersøgelse II. Række*, 79, 1–355. doi:10.34194/raekke2.v79.6868

Stiles, H. E. (1961). The association factor in information retrieval. *Journal of the ACM, 8*(2), 271–279. doi:10.1145/321062.321074

Tanimoto, T. T. (1958). An elementary mathematical theory of classification and prediction (PB167360). International Business Machines Corp., New York, NY.

Tarwid, K. (1960). Szacowanie zbieinosci nisz ekologicznych gatunkow droga oceny prawdopodobienstwa spotykania sie ich w polowach. *Ekologia Polska, 6*, 115–130.

Tulloss, R. E. (1997). Assessment of similarity indices for undesirable properties and a new tripartite similarity index based on cost functions. In M. E. Palm & I. H. Chapela (Eds.), *Mycology in sustainable development: Expanding concepts, vanishing borders* (pp. 122–143). Parkway Pub.

Walker, H. M., & Lev, J. (1953). *Statistical inference*. Holt.

Warrens, M. J. (2008). Similarity coefficients for binary data: Properties of coefficients, coefficient matrices, multi-way metrics and multivariate coefficients [Doctoral dissertation, Universiteit Leiden]. https://hdl.handle.net/1887/12987

Yule, G. U. (1900). On the association of attributes in statistics: With illustrations from the material of the childhood society, &c. *Philosophical Transactions of the Royal Society of London, 194*, 257–319. doi:10.1098/rsta.1900.0019

Yule, G. U. (1912). On the methods of measuring association between two attributes. *Journal of the Royal Statistical Society, 75*(6), 579–652. doi:10.2307/2340126

---

## `es_cohen_d` *Cohen d (for one-way ANOVA)*

---

### Description

An effect size measure for a one-way ANOVA. It simply compares the largest possible difference between two categories means and divides this over the total variance.

Note that most often Cohen d is reported with pairwise tests, but that is actually Cohen d_z. That version is available using es_cohen_d_ps().

### Usage

```
es_cohen_d(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used is (Cohen, 1988, p. 276):

$$d = \frac{\bar{x}_{max} - \bar{x}_{min}}{\sigma}$$

With:

$$\sigma = \sqrt{\frac{SS_w}{n}}$$

$$SS_w = \sum_{j=1}^{k}\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols*

- $x_{i,j}$ the i-th score in category j
- $n_j$ the number of scores in category j
- $k$ the number of categories
- $\bar{x}_j$ the mean of the scores in category j
- $SS_w$ the within sum of squares (sum of squared deviation of the mean)

## Value

the Cohen d value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

---

## `es_cohen_d_os` *Cohen d' (for one-sample)*

---

### Description

This function will calculate Cohen d' (one-sample). An effect size measure that can be used with a test for a single mean (for example a one-sample Student t-test).

### Usage

```
es_cohen_d_os(data, mu = NULL)
```

### Arguments

| | |
|---|---|
| `data` | pandas series with the numeric scores |
| `mu` | optional parameter to set the hypothesized mean. If not used the midrange is used |

### Details

The test is also described at PeterStatistics.com

The formula used (Cohen, 1988, p. 46):

$$d' = \frac{\bar{x} - \mu_{H_0}}{s}$$

With:

$$s = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n - 1}}$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

*Symbols used:*

- $\bar{x}$ the sample mean
- $\mu_{H_0}$ the hypothesized mean in the population
- $n$ the sample size (i.e. the number of scores)
- $s$ the unbiased sample standard deviation
- $x_i$ the i-th score

Note to use a rule-of-thumb from Cohen d, first convert this to a regular Cohen d using *es_convert(d', from="cohendos", to="cohend")*, then use *th_cohen_d(d)*

Or convert it further to an Odds Ratio using, *es_convert(d, from="cohend", to="or", ex1="chinn")* or *es_convert(d, from="cohend", to="or", ex1="borenstein")*. Then use *th_odds_ratio(or)*

### Value

Cohen d'. mu is also printed if not provided.

### Before, After and Alternatives

Before this you might want to perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z-Test.

After this you might want a rule-of-thumb for the effect size, first convert to regular Cohen d: `es_convert`, to convert Cohen's d one-sample to Cohen d, use *fr = "cohendos"* and *to = "cohend"*. `th_cohen_d`, for rules-of-thumb for Cohen d.

Alternative Effect Sizes: `es_hedges_g_os`, for Hedges g. `es_common_language_os`, for the Common Language Effect Size.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

### Examples

```
#Example 1: Numeric dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2['Gen_Age']
es_cohen_d_os(ex1)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
es_cohen_d_os(ex2)
```

---

## `es_cohen_d_ps` *Cohen d_z (for Paired Samples)*

### Description

An effect size measure for paired samples.

### Usage

```
es_cohen_d_ps(field1, field2, within = TRUE)
```

### Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `within` | boolean to use a correction for correlated pairs |

## Details

the formula used (Cohen, 1988, p. 48):

$$d_z = \frac{\bar{d}}{s_d}$$

With:

$$s_d = \sqrt{\frac{\sum_{i=1}^n \left(d_i - \bar{d}\right)^2}{n-1}}$$

$$d_i = x_{i,1} - x_{i,2}$$

$$\bar{d} = \frac{\sum_{i=1}^n d_i}{n}$$

### Symbols used:

- $n$ the number of pairs (sample size)
- $x_{i,1}$ the i-th score of the first variable
- $x_{i,2}$ the i-th score of the second variable

If *within=TRUE* the formula is changed to (Borenstein et al., 2009, p. 29):

$$d_z = \frac{\bar{d}}{s_w}$$

With:

$$s_w = \frac{s_d}{\sqrt{2 \times (1 - r_p)}}$$

$$r_p = \frac{\sum_{i=1}^n (x_{i,1} - \bar{x}_1) \times (x_{i,2} - \bar{x}_2)}{(n-1) \times s_1 \times s_2}$$

$$s_i^2 = \frac{\sum_{j=1}^n (x_{i,j} - \bar{x}_i)^2}{n_i - 1}$$

$$\bar{x}_i = \frac{\sum_{j=1}^n x_{i,j}}{n_i}$$

### Alternatives

library(lsr)

cohensD(var1, var2, method="paired")

*library(effsize)*

datF = na.omit(data.frame(var1, var2))

cohen.d(datF$var1, datF$var2, paired=TRUE)

cohen.d(datF$var1, datF$var2, paired=TRUE, within=FALSE)

## Value

the Cohen d effect size

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Borenstein, M., Hedges, L. V., Higgins, J. P. T., & Rothstein, H. R. (2009). Effect sizes based on means. In *Introduction to Meta-Analysis* (pp. 21–32). John Wiley & Sons, Ltd. doi:10.1002/9780470743386

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

---

# `es_cohen_f` *Cohen f*

## Description

An effect size measure for regression analysis or an ANOVA test. It gives roughly the proportion of variance explained by the categorical variable.

The Cohen f is often used with ANOVA, while Cohen f-squared with regression.

## Usage

```
es_cohen_f(nomField, scaleField, categories = NULL, useRanks = FALSE)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |
| `useRanks` | boolean, optional. Use of ranks or original scores. Default is FALSE |

## Details

The formula used (Cohen, 1988, p. 284):

$$f = \sqrt{\frac{\eta^2}{1 - \eta^2}}$$

Where $\eta^2$ is the value of eta-squared.

It can also be calculated using (Cohen, 1988, p. 371):

$$f = \frac{\sigma_\mu}{\sigma}$$

With:

$$\sigma_\mu = \sqrt{\frac{SS_b}{n}}$$

$$\sigma = \sqrt{\frac{SS_w}{n}}$$

Where $SS_i$ is the sum of squared differences, see the Fisher one-way ANOVA for details on how to calculate these.

The $f^2$ can be found in Cohen (1988, p. 410).

### Conversions

Cohen f can be converted to eta-squared using: *es_convert(f, from="cohenf", to="etasq")*

### Alternatives

*library(effectsize)*

anova_stats(aov(scores~groups))

cohens_f(aov(scores~groups))

## Value

the Cohen f value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

---

# `es_cohen_g` *Cohen's g*

## Description

Cohen's g (Cohen, 1988) is an effect size measure that could be accompanying a one-sample binomial (see Rosnow & Rosenthal, 2003), score or Wald test. It is simply the difference of the sample proportion with 0.5.

A video explanation of Cohen g can be found here on YouTube

This function is shown in this YouTube video and the effect size is also described at PeterStatistics.com

## Usage

```
es_cohen_g(data, p0Cat = NULL, codes = NULL)
```

## Arguments

| | |
|---|---|
| `data` | vector with the data |
| `p0Cat` | Optional the category for which p0=0.5 was used |
| `codes` | Optional vector with the two codes to use |

## Details

To decide on which category is associated with p0 the following is used:

- If codes are provided, the first code is assumed to be the category for the p0.
- If p0Cat is specified that will be used for p0 and all other categories will be considered as category 2, this means if there are more than two categories the remaining two or more (besides p0Cat) will be merged as one large category.
- If neither codes or p0Cat is specified and more than two categories are in the data a warning is printed and no results.
- If neither codes or p0Cat is specified and there are two categories, p0 is assumed to be for the category closest matching the p0 value (i.e. if p0 is above 0.5 the category with the highest count is assumed to be used for p0)

The formula used is (Cohen, 1988, p. 147):

$$g = p - 0.5$$

*Symbols used*:

- $p$ is the sample proportion

## Value

Dataframe with:

| | |
|---|---|
| `g for cat 1` | Cohen g for category 1 |
| `g for cat 2` | Cohen g for category 2 |

## Before, After and Alternatives

Before this effect size you might first want to perform a test: `ts_binomial_os`, for One-Sample Binomial Test `ts_score_os`, for One-Sample Score Test `ts_wald_os`, for One-Sample Wald Test

After this, you might want a rule-of-thumb: `th_cohen_g`, for rules-of-thumb for Cohen g

Alternatives for this effect size could be: `es_cohen_h_os`, for Cohen h' `es_alt_ratio`, for Alternative Ratio `r_rosenthal`, for Rosenthal Correlation if a z-value is available

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

## Examples

```
#Example 1: Numeric list
ex1 = c(1, 1, 2, 1, 2, 1, 2, 1)
es_cohen_g(ex1)

#Example 2: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
```

```
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
es_cohen_g(df1['sex'])
#Using two specific categories:
es_cohen_g(df1['mar1'], codes=c("DIVORCED", "NEVER MARRIED"))
```

---

## `es_cohen_h` *Cohen h*

### Description

Cohen h

### Usage

```
es_cohen_h(p1, p2)
```

### Arguments

| | |
|---|---|
| `p1` | the first proportion |
| `p2` | the second proportion |

### Details

Formula used (Cohen, 1988, p. 181):

$$h = \phi_1 - \phi_2$$

With:

$$\phi_i = 2 \times \text{arcsin}\sqrt{p_i}$$

*Symbols used*:

- $p_i$ the proportion of cases in category i

For classification rule-of-thumb use: *th_cohen_h()*

### Value

Cohen h

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

### Examples

```
es_cohen_h(0.2, 0.4)
```

---

`es_cohen_h_os` *Cohen's h'*

---

## Description

An adaptation of Cohen h (`es_cohen_h`) for a one-sample case. It is an effect size measure that could be accompanying a one-sample binomial, score or Wald test.

A YouTube video on Cohen h'.

This function is shown in this YouTube video and the effect size is also described at PeterStatistics.com

## Usage

```
es_cohen_h_os(data, p0 = 0.5, p0Cat = NULL, codes = NULL)
```

## Arguments

| | |
|---|---|
| `data` | a vector with the data |
| `p0` | Optional hypothesized proportion for the first category (default is 0.5) |
| `p0Cat` | Optional the category for which p0 was used |
| `codes` | Optional vector with the two codes to use |

## Details

To decide on which category is associated with p0 the following is used:

- If codes are provided, the first code is assumed to be the category for the p0.
- If p0Cat is specified that will be used for p0 and all other categories will be considered as category 2, this means if there are more than two categories the remaining two or more (besides p0Cat) will be merged as one large category.
- If neither codes or p0Cat is specified and more than two categories are in the data a warning is printed and no results.
- If neither codes or p0Cat is specified and there are two categories, p0 is assumed to be for the category closest matching the p0 value (i.e. if p0 is above 0.5 the category with the highest count is assumed to be used for p0)

Formula used (Cohen, 1988, p. 202):

$$h' = \phi_1 - \phi_{h_0}$$

With:

$$\phi_i = 2 \times \text{arcsin}\sqrt{p_i}$$

$$p_i = \frac{F_i}{n}$$

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

*Symbols used*:

- $F_i$ is the (absolute) frequency (count) of category i
- $n$ is the sample size, i.e. the sum of all frequencies
- $p_i$ the proportion of cases in category i
- $p_{h_0}$ the expected proportion (i.e. the proportion according to the null hypothesis)

## Value

Dataframe with:

| | |
|---|---|
| `Cohen h'` | the Cohen h' value |
| `comment` | the category for which p0 was |

## Before, After and Alternatives

Before this effect size you might first want to perform a test: `ts_binomial_os`, for One-Sample Binomial Test `ts_score_os`, for One-Sample Score Test `ts_wald_os`, for One-Sample Wald Test

After this, you might want a rule-of-thumb or first convert this to a 'regular' Cohen h: `es_convert`, to convert Cohen h' to Cohen h, use fr="cohenhos" and to=cohenh `th_cohen_h`, for rules-of-thumb for Cohen h

Alternatives for this effect size could be: `es_cohen_g`, for Cohen g `es_alt_ratio`, for Alternative Ratio `r_rosenthal`, for Rosenthal Correlation if a z-value is available

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

## Examples

```
#Example 1: Numeric list
ex1 = c(1, 1, 2, 1, 2, 1, 2, 1)
es_cohen_h_os(ex1)
es_cohen_h_os(ex1, p0=0.3)

#Example 2: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
es_cohen_h_os(df1['sex'])
```

---

## `es_cohen_kappa` *Cohen Kappa*

---

### Description

Cohen Kappa

### Usage

```
es_cohen_kappa(nom1, nom2, ase = "exact")
```

### Arguments

| | |
|---|---|
| `nom1` | the scores on the first variable |
| `nom2` | the scores on the second variable |
| `ase` | c("exact", "approximate") optional to indicate which method to use to calculate asymptotic standard errors |

### Details

The formula used is (Cohen, 1960, p. 40):

$$\kappa = \frac{p_0 - p_c}{1 - p_c}$$

With:

$$p_0 = \frac{P}{n}$$

$$p_c = \frac{Q}{n^2}$$

$$P = \sum_{i=1}^{r} F_{i,i}$$

$$Q = \sum_{i=1}^{r} R_i \times C_i$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

*Symbols used*

- $r$ is the number of rows (categories in the first variable)
- $c$ is the number of columns (categories in the second variable)
- $n$ is the total number of scores
- $F_{i,j}$ is the frequency (count) of scores equal to the i-th category in the first variable, and the j-th category in the second.

The approximate asymptotic standard errors (ase="approximate") use (Cohen, 1960, pp. 43-44):

$$ASE_0 \approx \sqrt{\frac{p_c}{n \times (1 - p_c)}}$$

$$ASE_1 \approx \sqrt{\frac{p_0 \times (1 - p_0)}{n \times (1 - p_c)^2}}$$

The exact asymptotic standard errors (ase="exact") use (Fleiss et al., 1969, p. 325):

$$ASE_0 = \sqrt{\frac{SS_0}{n \times (1 - p_c)^2}}$$

$$ASE_1 = \sqrt{\frac{SS_1}{n \times (1 - p_c)^4}}$$

With:

$$SS_0 = \left(\sum_{i=1}^{r} p_{i,.} \times p_{.,i} \times \left(1 - \left(p_{.,i} + p_{i,.}\right)\right)^2\right) - p_c^2 + (1 - p_0)^2 \times \sum_{i=1}^{r} \sum_{\substack{j=1 \\ j \neq i}}^{c} p_{i,.} \times p_{.,j} \times \left(p_{.,i} + p_{j,.}\right)^2$$

$$SS_1 = \left(\sum_{i=1}^{r} p_{i,i} \times \left((1 - p_c) - \left(p_{.,i} + p_{i,.}\right) \times (1 - p_0)\right)^2\right) - \left(p_0 \times p_c - 2 \times p_c + p_0\right)^2 + (1 - p_0)^2 \times \sum_{i=1}^{r} \sum_{\substack{j=1 \\ j \neq i}}^{c} p_{i,j} \times (p_{.,i}$$

$$p_{i,j} = \frac{F_{i,j}}{n}$$

$$p_{i,.} = \frac{R_i}{n}$$

$$p_{.,j} = \frac{C_j}{n}$$

The test is then performed using (Cohen, 1960, p. 44):

$$z_{\kappa} = \frac{\kappa}{ASE_0}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_{\kappa}\right|\right)\right)$$

Where $\Phi(\ldots)$ is the cumulative density function of the standard normal distribution

## Value

dataframe with the effect size value, the asymptotic standard errors (assuming null and alternative), test statistic, degrees of freedom, and p-value (sig.)

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Fleiss, J. L., Cohen, J., & Everitt, B. S. (1969). Large sample standard errors of kappa and weighted kappa. *Psychological Bulletin, 72*(5), 323–327. https://doi.org/10.1037/h0028106

---

## `es_cohen_u` *Cohen U*

---

### Description

Cohen (1988, p. 23) provided three measures that relate to Cohen's d.

- $U_1$, is the proportion of non-overlap between distributions
- $U_2$, is the proportion of overlap between distributions
- $U_3$, is the proportion of one group's scores below the mean of another group

$U_1$ and $U_2$ are probably the least used of these three.

By converting each back to Cohen's d, the rule-of-thumb from Cohen d could be used as classification. A nice interactive visualisation of the relation between Cohen $U_3$ and the Common Language Effect size, can be found at https://rpsychologist.com/therapist-effects/.

### Usage

```
es_cohen_u(d, version = "u3")
```

### Arguments

| | |
|---|---|
| `d` | the Cohen d value |
| `version` | "u3", "u2", "u1", Optional, the version of Cohen U to determine |

### Details

The following formulas are used (Cohen, 1988, p. 23):

$$U_3 = \Phi\left(d\right)$$

$$U_2 = \Phi\left(\frac{d}{2}\right)$$

$$U_1 = \Phi\left(\frac{2\times U_2 - 1}{U_2}\right)$$

*Symbols used:*

- $d$, Cohen's d value
- $n_i$ the number of scores in category i
- $\Phi\left(\ldots\right)$ the cumulative density function of the standard normal distribution

### Value

The Cohen U value

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

## See Also

`es_convert`, to convert an U to Cohen d use `fr="cohenu.", to="cohend"`.

---

## `es_cohen_w` *Cohen's w*

## Description

An effect size measure that could be used with a chi-square test. It has no upper limit, but can be compared to Cohen's rules-of-thumb.

This function is shown in this `YouTubevideo` and the measure is also described at PeterStatistics.com

## Usage

```
es_cohen_w(chi2, n)
```

## Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |

## Details

The formula used is (Cohen, 1988, p. 216):

$$w = \sqrt{\frac{\chi_{GoF}^2}{n}}$$

*Symbols used*:

- $\chi_{GoF}^2$ the Pearson chi-square goodness-of-fit value
- $n$ the sample size, i.e. the sum of all frequencies

## Value

value of Cohen's w

### Before, After and Alternatives

Before this you will need a chi-square value. From either: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test. `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

After this you might want to use some rule-of-thumb for the interpretation: `th_cohen_w` for various rules-of-thumb for Cohen w.

Alternative effect sizes that use a chi-square value: `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

### Examples

```
chi2Value <- 3.106
n <- 19
es_cohen_w(chi2Value, n)
```

---

## `es_common_language_is` *Common Language (CL/CLES) (Independent Samples)*

### Description

The Common Language Effect Size (a.k.a. Probability of Superiority) is the probability of taking a random pair from two categories, the first is greater than the first, i.e.

$$P(X > Y)$$

Note however that Wolfe and Hogg (1971) actually had this in reverse, i.e.

$$P(X \leq Y)$$

Some will also argue to count ties equally to each of the two categories (Grissom, 1994, p. 282), which makes the definition:

$$P(X > Y) + \frac{P(X = Y)}{2}$$

It was further developed by Vargha and Delaney (2000) especially in light of a Mann-Whitney U test.

For scale data, an approximation using the standard normal distribution is also available.

The term Common Language Effect Size can be found in McGraw and Wong (1992), the term Probability of Superiority is found in Grissom (1994), and the term Stochastic Superiority in Vargha and Delaney (2000)

## Usage

```
es_common_language_is(
  catField,
  scores,
  categories = NULL,
  levels = NULL,
  dmu = 0,
  method = "brute"
)
```

## Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scores` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `levels` | Optional list with the ordinal text values in order |
| `dmu` | Optional difference according to null hypothesis (default is 0) |
| `method` | Optional optional method to use. "brute" will use a brute force " that will split ties evenly, "brute-it" is the same as brute but ignores ties, "vda" will use the calculation from Vargha-Delany, and "appr" a normal approximation from McGraw-Wong |

## Details

For "brute" simply all possible pairs are determined and half of the ties are added, i.e. (Grissom, 1994, p. 282):

$$P(X > Y) + \frac{P(X = Y)}{2}$$

With "brute-it" the ties are ignored (it = ignore ties):

$$P(X > Y)$$

The "appr" uses the approximation from McGraw and Wong (1992, p. 361):

$$CL = \Phi\left(z\right)$$

With:

$$z = \frac{\left|\bar{x}_1 - \bar{x}_2\right|}{\sqrt{s_1^2 + s_2^2}}$$

$$s_i^2 = \frac{\sum_{j=1}^{n_i}\left(x_{i,j} - \bar{x}_i\right)^2}{n_i - 1}$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{i,j}}{n_i}$$

*Symbols used:*

- $x_{i,j}$ the j-th score in category i
- $n_i$ the number of scores in category i
- $\Phi\left(\dots\right)$ the cumulative density function of the standard normal distribution

The "vda" uses the formula used from Vargha and Delaney (2000, p. 107):

$$A = \frac{1}{n_j} \times \left(\frac{R_i}{n_i} - \frac{n_i + 1}{2}\right)$$

with $R_i$ the sum of the ranks in category i

It could also be calculated from the Mann-Whitney U value:

$$A = \frac{U}{n_1 \times n_2}$$

Note that the difference between the two options (using category 1 or category 2) will be the deviation from 0.5. If all scores in the first category are lower than the scores in the second, A will be 0 using the first category, and 1 for the second.

If the number of scores in the first category higher than the second, is the same as the other way around, A (no matter which category used) will be 0.5.

The CLE can be converted to a Rank Biserial (= Cliff delta) using the **es_convert()** function. This can then be converted to a Cohen d, and then the rules-of-thumb for Cohen d could be used (**th_cohen_d**())

The CLE for the other category is simply 1 - CLE, except for the case where ties are ignored ("brute-it").

## Value

A dataframe with:

| | |
|---|---|
| `CLE cat. 1` | the effect size for the first category |
| `CLE cat. 2` | the effect size for the second category |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Grissom, R. J. (1994). Statistical analysis of ordinal categorical status after therapies. *Journal of Consulting and Clinical Psychology, 62*(2), 281–284. doi:10.1037/0022-006X.62.2.281

McGraw, K. O., & Wong, S. P. (1992). A common language effect size statistic. *Psychological Bulletin, 111*(2), 361–365. doi:10.1037/0033-2909.111.2.361

Vargha, A., & Delaney, H. D. (2000). A critique and improvement of the CL common language effect size statistics of McGraw and Wong. *Journal of Educational and Behavioral Statistics, 25*(2), 101–132. doi:10.3102/10769986025002101

Wolfe, D. A., & Hogg, R. V. (1971). On constructing statistics and reporting data. *The American Statistician, 25*(4), 27–30. doi:10.1080/00031305.1971.10477278

## See Also

`th_cle`, to find rules-of-thumb for the CLE

## `es_common_language_os` *Common Language Effect Size (One-Sample)*

### Description

The Common Language Effect Size is most often used for independent samples or paired samples, but some have adapted the concept for one-sample as well.

It is the probability of taking a random score and the probability it is higher than the selected value: $$P(X > \mu_H_0)$$

Some will also argue to count ties equally, which makes the definition: $$P(X > \mu_H_0) + \fracP(X = \mu_H_0)2$$

This version is implemented in MatLab (see https://nl.mathworks.com/matlabcentral/fileexchange/113020-cles) based on a Python version from Tulimieri (2021)

For scale data, an approximation using the standard normal distribution is also available using Cohen's d, alternatively a conversion via the rank-biserial coefficient can be done. These two are used in R's *effectsize* library from Ben-Shachar et al. (2020).

The measure is also described at PeterStatistics.com

### Usage

```
es_common_language_os(scores, levels = NULL, mu = NULL, version = "brute")
```

### Arguments

| | |
|---|---|
| `scores` | list with scores as numbers, or if text also provide levels |
| `levels` | optional vector with levels in order |
| `mu` | optional hypothesized statistic, otherwise the midrange will be used |
| `version` | optional "brute", "brute-it", "rb", "normal" method to use. see details |

### Details

For "brute" simply counts all scores above the test statistic and half of the ones that are equal (Tulimieri, 2021): $$CL = P(X > \mu_H_0) + \fracP(X = \mu_H_0)2$$ With: $$P\left(x \gt \mu\right) = \frac\sum_i=1^n \begincases 1, & \textif x_i \gt \mu \\ 0, & \textotherwise\endcasesn$$ $$P\left(x = \mu\right) = \frac\sum_i=1^n \begincases 1, & \textif x_i = \mu \\ 0, & \textotherwise\endcasesn$$

This seems to also produce the same result as what Mangiafico (2016, pp. 223–224) calls a VDA-like measure, where VDA is short for Vargha-Delaney A.

With "brute-it" the ties are ignored (it = ignore ties): $$CL = P(X > \mu_H_0)$$

The "normal", uses Cohen's d and a normal approximation (Ben-Shachar et al., 2020): $$CL = \Phi\left(\fracd'\sqrt2\right)$$ Where \(d'\) is Cohen's d for one-sample, and \(\Phi\left(\dots\right)\) the cumulative density function of the normal distribution This is like a one-sample version of the McGraw and Wong (1992, p. 361) version with the independent samples.

The "rb", uses the rank-biserial correlation coefficient (Ben-Shachar et al., 2020): $$CL = \frac1+r_b2$$ The CLE can be converted to a Rank Biserial (= Cliff delta) using the **es_convert()** function. This can then be converted to a Cohen d, and then the rules-of-thumb for Cohen d could be used (**th_cohen_d**())

### Value

CLES : float, the Common Language Effect Size

### Before, After and Alternatives

Before this measure you might want to perform the test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test. `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z-Test.

After this you might want a rule-of-thumb directly or by converting this to either rank biserial or Cohen d: `th_cle`, for CLES rule-of-thumb (incl. conversion options)

Alternative effect size measure with ordinal data: `es_dominance`, for the Dominance score. `r_rank_biserial_os`, for the Rank-Biserial Correlation `r_rosenthal`, for the Rank-Biserial Correlation

Alternative effect size measure with interval or ratio data: `es_cohen_d_os`, for for Cohen d'. `es_hedges_g_os`, for Hedges g.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Ben-Shachar, M., Lüdecke, D., & Makowski, D. (2020). effectsize: Estimation of Effect Size Indices and Standardized Parameters. *Journal of Open Source Software, 5*(56), 1–7. doi:10.21105/joss.02815

Grissom, R. J. (1994). Statistical analysis of ordinal categorical status after therapies. *Journal of Consulting and Clinical Psychology, 62*(2), 281–284. doi:10.1037/0022-006X.62.2.281

Mangiafico, S. S. (2016). *Summary and analysis of extension program evaluation in R* (1.20.01). Rutger Cooperative Extension.

McGraw, K. O., & Wong, S. P. (1992). A common language effect size statistic. *Psychological Bulletin, 111*(2), 361–365. doi:10.1037/0033-2909.111.2.361

Tulimieri, D. (2021). CLES/CLES. https://github.com/tulimid1/CLES/tree/main

Wolfe, D. A., & Hogg, R. V. (1971). On constructing statistics and reporting data. *The American Statistician, 25*(4), 27–30. doi:10.1080/00031305.1971.10477278

### Examples

```
# Example 1: Dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = studentDf[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
es_common_language_os(ex1, levels=order)

# Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
es_common_language_os(ex2)

# Example 3: Text data with
ex3 = c("a", "b", "f", "d", "e", "c")
order = c("a", "b", "c", "d", "e", "f")
es_common_language_os(ex3, levels=order)
```

---

## `es_common_language_ps` *Common Language Effect Size (Paired-Samples)*

---

### Description

the probability that a randomly selected score from the one population will be greater than a randomly sampled score from the other population.

### Usage

```
es_common_language_ps(
  field1,
  field2,
  dmu = 0,
  method = c("dunlap", "mcgraw-wong")
)
```

### Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `dmu` | difference according to null hypothesis (default is 0), only if method="mcgraw-wong" |
| `method` | method to use for calculating CL (see details) |

### Value

cl, float. the common language effect size measure value

The formula used (McGraw & Wong, 1992, p. 363):

$$CL = \Phi\left(z_{cl}\right)$$

With:

$$z_{cl} = \frac{\left|\bar{x}_1 - \bar{x}_2\right| - d_{H0}}{\sqrt{s_1^2 + s_2^2 - 2\times r_p \times s_1 \times s_2}}$$

$$s_i^2 = \frac{\sum_{j=1}^{n}\left(x_{i,j} - \bar{x}_i\right)^2}{n_i - 1}$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n} x_{i,j}}{n_i}$$

$$r_p = \frac{\sum_{i=1}^{n}\left(x_{i,1} - \bar{x}_1\right)\times\left(x_{i,2} - \bar{x}_2\right)}{\left(n-1\right)\times s_1 \times s_2}$$

**Symbols used:**

- $n$ the total number of pairs
- $x_{i,j}$ the i-th score in the j-th variable
- $r_p$ the Pearson correlation coefficient

This equation is used when method="mcgraw-wong"

The formula used for the Dunlap method (Dunlap, 1994, p. 509):

$$CL = \sin^{-1}(r) + \frac{1}{2}$$

This equation is used when method="dunlap".

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Dunlap, W. P. (1994). Generalizing the common language effect size indicator to bivariate normal correlations. *Psychological Bulletin, 116*(3), 509–511. https://doi.org/10.1037/0033-2909.116.3.509

McGraw, K. O., & Wong, S. P. (1992). A common language effect size statistic. *Psychological Bulletin, 111*(2), 361–365. https://doi.org/10.1037/0033-2909.111.2.361

---

## `es_cont_coeff` *(Pearson) Contingency Coefficient*

### Description

(Pearson) Contingency Coefficient

### Usage

```
es_cont_coeff(chi2, n, adj = NULL, r = NULL, c = NULL)
```

### Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |
| `adj` | c(NULL, "sakoda") adjustment to use (see details) |
| `r` | the number of rows, required if adj="sakado" |
| `c` | the number of columns, required if adj="sakado" |

### Details

The formula used is (Pearson, 1904, p. 9):

$$C = \sqrt{\frac{\chi^2}{n + \chi^2}}$$

*Symbols used*:

- $\chi^2$ the chi-square test statistic
- $n$ the sample size, i.e. the sum of all frequencies

The maximum value for C would be (Sakoda, 1977, p. 778):

$$C_{max} = \sqrt{\frac{m-1}{m}}$$

Where $m$ is the minimum of the number of rows, or number of columns.

Sakoda propses to divide the contingency coefficient by this maximum, i.e.

$$\frac{C}{C_{max}}$$

For a 2x2 table Cole C1 will also divide by the maximum and produce the same result.

Blaikie-Roberts suggest to use as $C_{max}$ (Blaikie, 1969, p.19):

$$C_{max} = \sqrt[4]{\frac{r-1}{r} \times \frac{c-1}{c}}$$

Blaikie refers to his mentor Roberts for this (Blaikie, 2003, p. 115)

**Alternative**

The *'DescTools'* library also has a function for this: *ContCoef()*

## Value

value of (Pearson) Contingency Coefficient

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Blaikie, N. W. H. (1969). Religion, social status, and community involvement: A study in Christchurch. *The Australian and New Zealand Journal of Sociology, 5*(1), 14–31. doi:10.1177/144078336900500102

Blaikie, N. W. H. (2003). *Analyzing quantitative data: From description to explanation.* Sage Publications Ltd.

Pearson, K. (1904). *Contributions to the Mathematical Theory of Evolution. XIII. On the theory of contingency and its relation to association and normal correlation.* Dulau and Co.

Sakoda, J. M. (1977). Measures of Association for Multivariate Contingency Tables. *In Proceedings of the Social Statistics Section of the American Statistical Association: Vol. Part III* (pp. 777–780).

## Examples

```
chi2Value <- 3.105263
n <- 19
es_cont_coeff(chi2Value, n)
```

---

`es_convert` *Convert Effect Size*

---

## Description

Convert Effect Size

## Usage

```
es_convert(es, fr, to, ex1 = NULL, ex2 = NULL)
```

## Arguments

| | |
|---|---|
| `es` | the effect size value to convert |
| `fr` | name of the original effect size (see details) |
| `to` | name of the effect size to convert to (see details) |
| `ex1` | extra for some conversions (see details) |
| `ex2` | extra for some conversions (see details) |

## Details

**COHEN D**

**Cohen d to Odds Ratio**

fr="cohend", to="or", ex1="chinn"

This uses (Chinn, 2000, p. 3129):

$$OR = e^{d\times 1.81}$$

fr="cohend", to="or", ex1="borenstein"

This uses (Borenstein et. al, 2009, p. 3):

$$OR = e^{\frac{d\times\pi}{\sqrt{3}}}$$

**Cohen d to Rank Biserial (Cliff delta)**

fr = "cohend", to = "rb"

This uses (Marfo & Okyere, 2019, p. 4):

$$r_b = \frac{2\times\Phi\left(\frac{d}{2}\right) - 1}{\Phi\left(\frac{d}{2}\right)}$$

**COHEN D'**

**Convert a Cohen d' to Cohen d**

fr="cohendos" to="cohend"

This uses (Cohen , 1988, p. 46):

$$d = d' \times \sqrt{2}$$

**COHEN F**

**Cohen f to Eta-squared**

fr="cohenf" to="etasq"

This uses (Cohen, 1988, p. 284):

$$\eta^2 = \frac{f^2}{1 + f^2}$$

**COHEN H'**

**Cohen h' to Cohen h**

fr = "cohenhos", to = "cohenh"

This uses (Cohen, 1988, p. 203):

$$h = h' \times \sqrt{2}$$

**COHEN U**

**Cohen U to Cohen d**

fr="cohenu1", to="cohend" fr="cohenu2", to="cohend" fr="cohenu3", to="cohend"

This uses (Cohen, 1988, p. 23):

$$d = \Phi^{-1}\left(U_3\right)$$

$$d = 2 \times \Phi^{-1}\left(U_2\right)$$

$$d = 2 \times \Phi^{-1}\left(\frac{1}{2 - U_1}\right)$$

**COHEN w**

**Cohen w to Contingency Coefficient**

fr="cohenw", to="cc"

*Cohen w to Cramér V GoF*

fr="cohenw", to="cramervgof", ex1=k

This uses (Cohen, 1988, p. 223):

$$v = \frac{w}{\sqrt{k - 1}}$$

*Cohen w to Cramér V ind.*

fr="cohenw", to="cramervind", ex1=r, ex2=c

This uses:

$$v = \frac{w}{\sqrt{\min\left(r - 1, c - 1\right)}}$$

*Cohen w to Fei*

fr="cohenw", to="fei", ex1=minExp/n

This uses:

$$Fei = \frac{w}{\sqrt{\frac{1}{p_E} - 1}}$$

**CRAMER V GoF**

**Cramer's v for Goodness-of-Fit to Cohen w**

fr="cramervgof", to = "cohenw", ex1 = k

This uses (Cohen, 1988, p. 223):

$$w = v \times \sqrt{df}$$

**EPSILON SQUARED**

**Epsilon Squared to Eta Squared**

fr="epsilonsq", to="etasq", ex1 = n, ex2 = k

This uses:

$$\eta^2 = 1 - \frac{\left(1-\epsilon^2\right) \times (n-k)}{n-1}$$

**Epsilon Squared to Omega Squared**

fr="epsilonsq", to="omegasq", ex1=MS_w, ex2 = SS_t

This uses:

$$\hat{\omega}^2 = \epsilon^2 \times \left(1 - \frac{MS_w}{SS_t + MS_w}\right)$$

**ETA SQUARED**

**Eta squared to Cohen f** fr="etasq", to="cohenf")

This uses:

$$f = \sqrt{\frac{\eta^2}{1-\eta^2}}$$

**Eta squared to Epsilon Squared** fr="etasq", to="epsilonsq", ex1=n, ex2=k

This uses:

$$\epsilon^2 = \frac{n \times \eta^2 - k + \left(1-\eta^2\right)}{n-l}$$

**FEI**

*Fei to Cohen w*

fr="fei", to="cohenw", ex1=minExp/n

This uses:

$$w = Fei \times \sqrt{\frac{1}{p_E} - 1}$$

*Fei to Johnston-Berry-Mielke E* fr="fei", to="jbme"

This uses:

$$E = Fei^2$$

**JOHNSTON-BERRY-MIELKE**

**Johnston-Berry-Mielke E to Cohen w**

fr="jbme", to="cohenw", ex1=minExp/n

This uses (Johnston et al., 2006, p. 413):

$$w = \sqrt{\frac{E \times (1-)}{q}}$$

*Johnston-Berry-Mielke E to Cohen w*

fr="jbme", to="fei"

This uses:

$$Fei = \sqrt(E)$$

**ODDS RATIO**

**Odds Ratio to Cohen d**

fr="or", to="cohend", ex1="chinn"

This uses (Chinn, 2000, p. 3129):

$$d = \frac{\ln(OR)}{1.81}$$

fr="or", to="cohend", ex1="borenstein"

This uses (Borenstein et. al, 2009, p. 3):

$$d = \ln(OR) \times \frac{\sqrt{3}}{\pi}$$

**Odds Ratio to Yule Q**

fr="or", to="yuleq"

This uses:

$$Q = \frac{OR - 1}{OR + 1}$$

**Odds Ratio to Yule Y**

This uses

$$Y = \frac{\sqrt{OR} - 1}{\sqrt{OR} + 1}$$

**OMEGA SQUARED**

**Omega Squared to Epsilon Squared**

fr="omegasq", to="epsilonsq", ex1=MS_w, ex2 = SS_t

This uses:

$$\epsilon^2 = \frac{\hat{\omega}^2}{1 - \frac{MS_w}{SS_t + MS_w}}$$

**RANK BISERIAL (CLIFF DELTA)**

**Rank Biserial (Cliff delta) to Cohen d**

fr = "rb", to = "cohend"

This uses (Marfo & Okyere, 2019, p. 4):

$$d = \sqrt{2} \times \Phi^{-1}\left(-\frac{1}{r_b - 2}\right)$$

**Rank Biserial (Cliff delta) to Vargha-Delaney A**

fr = "rb", to = "vda"

This uses:

$$r_b = 2 \times A - 1$$

**VARGHA-DELANEY A**

**Vargha-Delaney A to Rank Biserial (Cliff delta)**

fr = "vda", to = "rb"

This uses:

$$A = \frac{r_b + 1}{2}$$

**YULE Q**

**Yule Q to Odds Ratio**

fr="yuleq", to="or"

This uses:

$$OR = \frac{1 + Q}{1 - Q}$$

**Yule Q to Yule Y**

fr="yuleq", to="yuley"

This uses:

$$Y = \frac{1 - sqrt1 - Q^2}{Q}$$

**YULE Y**

**Yule Y to Yule Q**

fr="yuley", to=="yuleq"

This uses:

$$Q = \frac{2 \times Y}{1 + Y^2}$$

**Yule Y to Odds Ratio**

fr="yuley", to=="or"

This uses

$$OR = \left(\frac{1 + Y}{1 - Y}\right)^2$$

## Value

the converted effect size value

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Borenstein, M., Hedges, L. V., Higgins, J. P. T., & Rothstein, H. R. (2009). Converting Among Effect Sizes. In *Introduction to Meta-Analysis*. John Wiley & Sons, Ltd. https://doi.org/10.1002/9780470743386

Chinn, S. (2000). A simple method for converting an odds ratio to effect size for use in meta-analysis. *Statistics in Medicine, 19*(22), 3127–3131. https://doi.org/10.1002/1097-0258(20001130)19:22<3127::aid-sim784>3.0.co;2-m

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Johnston, J. E., Berry, K. J., & Mielke, P. W. (2006). Measures of effect size for chi-squared and likelihood-ratio goodness-of-fit tests. *Perceptual and Motor Skills, 103*(2), 412–414. https://doi.org/10.2466/pms.103.2.4 414

## Examples

```
es_convert(0.3, fr="cohenhos", to = "cohenh")
```

---

# `es_cramer_v_gof` *Cramer's V for Goodness-of-Fit*

## Description

Cramér's V is one possible effect size when using a chi-square test. This measure is actually designed for the chi-square test for independence but can be adjusted for the goodness-of-fit test (Kelley & Preacher, 2012, p. 145; Mangiafico, 2016, p. 474).

It gives an estimate of how well the data then fits the expected values, where 0 would indicate that they are exactly equal. If you use the equal distributed expected values the maximum value would be 1, otherwise it could actually also exceed 1.

As for a classification Cramér's V can be converted to Cohen w, for which Cohen provides rules of thumb.

A Bergsma correction is also possible.

A general explanation can also be found in this YouTube video. The function is shown in this `YouTubevideo` and the test is also described at PeterStatistics.com

## Usage

```
es_cramer_v_gof(chi2, n, k, bergsma = FALSE)
```

## Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |
| `k` | the number of categories |
| `bergsma` | optional boolean to indicate the use of the Bergsma correction: `FALSE`(the default), or `TRUE` |

## Details

The formula used is:

$$V = \sqrt{\frac{\chi_{GoF}^{2}}{n \times (k - 1)}}$$

*Symbols used*:

- $k$ the number of categories
- $n$ the sample size, i.e. the sum of all frequencies
- $\chi_{GoF}^{2}$ the chi-square value of a Goodness-of-Fit test

The Bergsma correction uses a different formula.

$$\tilde{V} = \sqrt{\frac{\tilde{\varphi}^2}{\tilde{k} - 1}}$$

With:

$$\tilde{\varphi}^2 = max\left(0, \varphi^2 - \frac{k-1}{n-1}\right)$$

$$\tilde{k} = k - \frac{\left(k-1\right)^2}{n-1}$$

$$\varphi^2 = \frac{\chi_{GoF}^2}{n}$$

Cramér described V (1946, p. 282) for use with a test of independence. Others (e.g. K. Kelley & Preacher, 2012, p. 145; Mangiafico, 2016a, p. 474) added that this can also be use for goodness-of-fit tests.

For the Bergsma (2013, pp. 324-325) correction the same thing applies

Cramér's V can be converted to Cohen's w using *es_convert(from="cramervgof", to = "cohenw", ex1 = df)*

Rules-of-thumb for the interpretation can then be used, using *th_cohen_w(w)*

## Value

Cramer's V value

## Alternatives

The *lsr* library has a similar function: *cramersV()*

The *DescTools* library has a similar function: *CramerV()*

## Before, After and Alternatives

Before this you will need a chi-square value. From either: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test. `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

After this you might want to use some rule-of-thumb for the interpretation: `th_cramer_v` for various rules-of-thumb for Cramér V.

or convert to Cohen w: `es_convert` to convert Cramér's V to Cohen w (using fr="cramervgof", to="cohenw", ex1=k). `th_cohen_w` for various rules-of-thumb for Cohen w.

Alternative effect sizes that use a chi-square value: `es_cohen_w`, for Cohen w. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Bergsma, W. (2013). A bias-correction for Cramér's and Tschuprow's. *Journal of the Korean Statistical Society, 42*(3), 323–328. https://doi.org/10.1016/j.jkss.2012.10.002

Cramér, H. (1946). *Mathematical methods of statistics*. Princeton University Press.

Kelley, K., & Preacher, K. J. (2012). On effect size. *Psychological Methods, 17*(2), 137–152. https://doi.org/10.1037/a0028086

Mangiafico, S. S. (2016). Summary and analysis of extension program evaluation in R (1.13.5). Rutger Cooperative Extension.

### Examples

```
chi2Value <- 3.106
n <- 19
k <- 3
es_cramer_v_gof(chi2Value, n, k)
es_cramer_v_gof(chi2Value, n, k, bergsma=TRUE)
```

---

## `es_cramer_v_ind` *Cramer's V for Independence Test*

### Description

Cramer's V for Independence Test

### Usage

```
es_cramer_v_ind(chi2, n, r, c, cc = NULL)
```

### Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |
| `r` | the number of categories in the first variable (i.e. the number of rows) |
| `c` | the number of categories in the second variable (i.e. the number of columns) |
| `cc` | c(NULL, "bergsma") optional to indicate correction to use (default is NULL) |

### Details

The formula used is:

$$V = \sqrt{\frac{\chi^2}{n \times (\min(r, c) - 1)}}$$

*Symbols used*:

- $r$ the number of categories in the first variable (i.e. the number of rows)
- $c$ the number of categories in the second variable (i.e. the number of columns)
- $n$ the sample size, i.e. the sum of all frequencies
- $\chi^2$ the chi-square statistic

The Bergsma correction uses a different formula (Bergsma, 2013, pp. 324-325):

$$V_B = \sqrt{\frac{\tilde{\varphi}^2}{\min\left(\tilde{r}, \tilde{c}\right) - 1}}$$

With:

$$\tilde{\varphi}^2 = \max\left(0, \varphi^2 - \frac{(r-1) \times (c-1)}{n-1}\right)$$

$$\tilde{r} = r - \frac{\left(r-1\right)^2}{n-1}$$

$$\tilde{c} = r - \frac{\left(c-1\right)^2}{n-1}$$

$$\varphi^2 = \frac{\chi^2}{n}$$

## Value

Cramer’s V value

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Bergsma, W. (2013). A bias-correction for Cramér’s and Tschuprow’s. Journal of the Korean Statistical Society, 42(3), 323–328. https://doi.org/10.1016/j.jkss.2012.10.002

Cramér, H. (1946). Mathematical methods of statistics. Princeton University Press.

## Examples

```
chi2Val = 16.98975
n = 1941
nRows = 5
nCols = 2
es_cramer_v_ind(chi2Val, n, nRows, nCols)
es_cramer_v_ind(chi2Val, n, nRows, nCols, cc="bergsma")
```

# `es_dominance` *Dominance and a Vargha-Delaney A like effect size measure*

## Description

This measure could be used with a sign test, since it does not rely on a z-value.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

## Usage

```
es_dominance(data, levels = NULL, mu = NULL, out = "dominance")
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores as numbers, or if text also provide levels |
| `levels` | optional vector with levels in order |
| `mu` | optional parameter to set the hypothesized median. If not used the midrange is used |
| `out` | c("dominance","vda"). optional to either show the dominance score or a VDA like measure: `"dominance"` (default), `"vda"` |

## Details

The formula used is (Mangiafico, 2016, p. 223-224):

$$D = p_{pos} - p_{neg}$$

Where:

$$p_i = \frac{n_i}{n}$$

*Symbols used:*

- $p_{pos}$ the proportion of cases above the hypothesized median
- $p_{neg}$ the proportion of cases below the hypothesized median
- $n_{pos}$ the number of cases above the hypothesized median
- $n_{neg}$ the number of cases below the hypothesized median
- $n$ the total number of cases

The dominance score will range from -1 to 1.

A Vargha-Delaney A (VDA) style effect size is calculated with (Mangiafico, 2016, p. 223-224):

$$VDA_{like} = \frac{D + 1}{2}$$

This will range from 0 to 1, with 0.5 being the same as a dominance score of 0.

## Value

dataframe with the hypothesized median (mu) and the effect size value

### Before, After and Alternatives

Before this measure you might want to perform the test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

Alternative effect size measure with ordinal data: `es_common_language_os`, for the Common Language Effect Size. `r_rank_biserial_os`, for the Rank-Biserial Correlation `r_rosenthal`, for the Rank-Biserial Correlation

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Mangiafico, S. S. (2016). Summary and analysis of extension program evaluation in R (1.20.01). Rutger Cooperative Extension.

### Examples

```
#Example 1: Text dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
es_dominance(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
es_dominance(ex2)
```

---

## `es_epsilon_sq` *Epsilon Squared*

### Description

An effect size measure to indicate the the strength of the categories on the ordinal/scale field. A 0 would indicate no influence, and 1 a perfect relationship.

This is an attempt to make eta-squared unbiased (applying a population correction ratio) (Kelley, 1935, p. 557). Although a popular belief is that omega-squared is preferred over epsilon-squared (Keselman, 1975), a later study actually showed that epsilon-squared might be preferred (Okada, 2013).

Tomczak and Tomczak (2014) recommend this as one option to be used with a Kruskal-Wallis test, however I think they labelled epsilon-squared as eta-squared and the other way around.

### Usage

```
es_epsilon_sq(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  useRanks = FALSE
)
```

### Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `useRanks` | boolean, optional. Use ranks or use the scores as given in ordfield. Default is FALSE. |

### Details

The formula used (Kelley, 1935, p. 557):

$$\epsilon^2 = \frac{n \times \eta^2 - k + \left(1 - \eta^2\right)}{n - k}$$

*Symbols used:*

- $\eta^2$ eta squared
- $k$ the number of categories
- $n$ the sample size

There are quite some variations on this formula.

For example Cureton (1966, p. 605):

$$\epsilon^2 = 1 - \frac{n-1}{n-k} \times \left(1 - \eta^2\right)$$

Caroll and Nordholm (1975, p. 547):

$$\epsilon^2 = \frac{F - 1}{F + \frac{n-k}{k-1}}$$

Albers and Lakens (2018, p. 194)

$$\epsilon^2 = \frac{F - 1}{F + \frac{df_w}{df_b}}$$

Albers and Lakens (2018, p. 188)

$$\epsilon^2 = \frac{SS_b - df_b \times MS_w}{SS_t}$$

**Conversions**

To convert $\epsilon^2$ to $\eta^2$ use *es_conver(epsilonsq, from="epsilonsq", to="etasq", ex1=n, ex2=k)*

To convert $\epsilon^2$ to $\omega^2$ use *es_convert(epsilonsq, from="etasq", to="omegasq", ex1=MS_w, ex2=SS_t)*

**Alternatives**

*library(effectsize)*

anova_stats(aov(scores~groups))

epsilon_squared(aov(scores~groups))

## Value

epsSq, float. The epsilon squared value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Albers, C., & Lakens, D. (2018). When power analyses based on pilot data are biased: Inaccurate effect size estimators and follow-up bias. *Journal of Experimental Social Psychology, 74*, 187–195. doi:10.1016/j.jesp.2017.09.004

Carroll, R. M., & Nordholm, L. A. (1975). Sampling characteristics of Kelley's $\epsilon$ and Hays' $\omega$. *Educational and Psychological Measurement, 35*(3), 541–554. doi:10.1177/001316447503500304

Cureton, E. E. (1966). On correlation coefficients. *Psychometrika, 31*(4), 605–607. doi:10.1007/BF02289528

Kelley, T. L. (1935). An unbiased correlation ratio measure. *Proceedings of the National Academy of Sciences, 21*(9), 554–559. doi:10.1073/pnas.21.9.554

Keselman, H. J. (1975). A Monte Carlo investigation of three estimates of treatment magnitude: Epsilon squared, eta squared, and omega squared. *Canadian Psychological Review / Psychologie Canadienne, 16*(1), 44–48. doi:10.1037/h0081789

Okada, K. (2013). Is omega squared less biased? A comparison of three major effect size indices in one-way anova. *Behaviormetrika, 40*(2), 129–147. doi:10.2333/bhmk.40.129

Pearson, K. (1911). On a correction to be made to the correlation ratio $\eta$. *Biometrika, 8*(1/2), 254. doi:10.2307/2331454

Tomczak, M., & Tomczak, E. (2014). The need to report effect size estimates revisited. An overview of some recommended measures of effect size. *Trends in Sport Sciences, 1*(21), 19–25.

---

# `es_eta_sq` *Eta Squared*

## Description

An effect size measure to indicate the the strength of the categories on the ordinal/scale field. A 0 would indicate no influence, and 1 a perfect relationship.

It is “the proportion of the variation in Y that is associated with membership of the different groups defined by X “ (Richardson, 2011, p. 136).

An alternative Epsilon Squared is an attempt to make eta-squared unbiased (applying a population correction ratio) (Kelley, 1935, p. 557). Although a popular belief is that omega-squared is preferred over epsilon-squared (Keselman, 1975), a later study actually showed that epsilon-squared might be preferred (Okada, 2013).

Tomczak and Tomczak (2014) recommend this this as one option to be used with a Kruskal-Wallis test, however I think they labelled epsilon-squared as eta-squared and the other way around.

### Usage

```
es_eta_sq(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  useRanks = FALSE
)
```

### Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `useRanks` | boolean, optional. Use ranks or use the scores as given in ordfield. Default is FALSE. |

### Details

The formula used is (Pearson, 1911, p. 254):

$$\eta^2 = \frac{SS_b}{SS_t}$$

With:

$$SS_t = \sum_{j=1}^{k}\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}\right)^2$$

$$SS_b = \sum_{j=1}^{k}\left(\bar{x}_j - \bar{x}\right)^2$$

$$\bar{x}_j = \frac{\sum_{j=1}^{n_j} x_{i,j}}{n_j}$$

$$\bar{x} = \sum_{j=1}^{k}\sum_{i=1}^{n_j} x_{i,j} n$$

$$n = \sum_{j=1}^{k} n_j$$

*Symbols used:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j

There are variations on the formula that will give the same result, for example:

$$\eta^2 = \frac{F \times (k-1)}{F \times (k-1) + n - k}$$

or

$$\eta^2 = \frac{F \times df_b}{F \times df_b + df_w}$$

If ranks are used, the eta-squared can also be determined using (Tomczak & Tomczak, 2014, p. 24):

$$\eta^2 = \frac{H}{n-1}$$

*Symbols used:*

- $n$, the total sample size
- $k$, the number of categories
- $SS_b$, the between sum of squares (sum of squared deviation of the mean)
- $SS_t$, the total sum of squares (sum of squared deviation of the mean)
- $F$, the F-statistic
- $H$, H-statistic from Kruskal-Wallis H-test
- $df_i$, the degrees of freedom of i
- $x_{i,j}$, the i-th score in category j
- $n_j$, the number of scores in category j
- $\bar{x}_j$, the mean of the scores in category j
- $b$, is between = factor = treatment = model
- $w$, is within = error (the variability within the groups)

Eta-squared can be converted to Cohen f, using *es_convert(etasq, from="etasq", to="cohenf")*

Eta-squared can be converted to Epsilon square, using *es_convert(etasq, from="etasq", to="epsilonsq", ex1=n, ex2=k)*

**Alternatives**

*library(lsr)*

etaSquared(aov(scores~groups))

*library(effectsize)*

anova_stats(aov(scores~groups))

eta_squared(aov(scores~groups))

## Value

etaSq, float. The eta squared value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Kelley, T. L. (1935). An unbiased correlation ratio measure. *Proceedings of the National Academy of Sciences, 21*(9), 554–559. doi:10.1073/pnas.21.9.554

Keselman, H. J. (1975). A Monte Carlo investigation of three estimates of treatment magnitude: Epsilon squared, eta squared, and omega squared. *Canadian Psychological Review / Psychologie Canadienne, 16*(1), 44–48. doi:10.1037/h0081789

Okada, K. (2013). Is omega squared less biased? A comparison of three major effect size indices in one-way anova. *Behaviormetrika, 40*(2), 129–147. doi:10.2333/bhmk.40.129

Pearson, K. (1911). On a correction to be made to the correlation ratio $\eta$. *Biometrika, 8*(1/2), 254. doi:10.2307/2331454

Richardson, J. T. E. (2011). Eta squared and partial eta squared as measures of effect size in educational research. *Educational Research Review, 6*(2), 135–147. doi:10.1016/j.edurev.2010.12.001

Tomczak, M., & Tomczak, E. (2014). The need to report effect size estimates revisited. An overview of some recommended measures of effect size. *Trends in Sport Sciences, 1*(21), 19–25.

---

# `es_fei` *Fei*

## Description

An effect size measure that could be used with a chi-square test or g-test.

## Usage

```
es_fei(chi2, n, minExp)
```

## Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |
| `minExp` | the minimum expected count |

## Details

The formula used (Ben-Shachar et al., 2023, p. 6):

$$Fei = sqrtfracchi_{GoF}^{2}ntimesleft(frac1minleft(p_{E}right) - 1right)$$

*Symbols used*

- $chi_{GoF}^{2}$, the chi-square value of the goodness-of-fit chi-square test
- $n$, the sample size
- $p_{E}$, the expected proportions

*Classification* A qualification rule-of-thumb could be obtained by converting this to Cohen's w (use **es_convert(Fei, fr="fei", to="cohenw", ex1=minExp/n)**)

### Value

the value of Fei

### Before, After and Alternatives

Before this you will need a chi-square value. From either: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test. `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

After this you might want to use some rule-of-thumb for the interpretation by converting it to Cohen w: `es_convert` to convert Fei to Cohen w (using fr="fei", to="cohenw", ex1=minExp/n). `th_cohen_w` for various rules-of-thumb for Cohen w.

Alternative effect sizes that use a chi-square value: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_jbm_e`, for Johnston-Berry-Mielke E.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Ben-Shachar, M. S., Patil, I., Thériault, R., Wiernik, B. M., & Lüdecke, D. (2023). Phi, fei, fo, fum: Effect sizes for categorical data that use the chi-squared statistic. *Mathematics, 11*(1982), 1–10. doi:10.3390/math11091982

### Examples

```
chi2 = 23.5
n = 53
minExp = 14
es_fei(chi2=chi2, n=n, minExp=minExp)
```

---

## `es_freeman_theta` *Freeman Theta*

### Description

According to Jacobson (1972, p. 42), this is the only measure for nominal-ordinal data, and is a modification of Somers d.

It can range from 0 to 1, with 0 indicating no influence of the catField on the scores of the ordField, and a 1 a perfect relationship.

Alternatives could be eta-squared and epsilon-squared.

### Usage

```
es_freeman_theta(catField, ordField, categories = NULL, levels = NULL)
```

## Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |

## Details

The formula used is (Freeman, 1965, p. 116):

$$\theta = \frac{D}{T}$$

With:

$$D = \sum D_{x,y}$$

$$D_{x,y} = |f_a - f_b|$$

$$f_a = \sum_{i=1}^{n_{lvl}-1} \left(F_{x,i} \times \sum_{j=i+1}^{n_{lvl}} F_{y,j}\right)$$

$$f_b = \sum_{i=2}^{n_{lvl}} \left(F_{x,i} \times \sum_{j=1}^{i-1} F_{y,j}\right)$$

*Symbols used:*

- $F_{x,i}$, from category x, the number of cases with level i.
- $n_{lvl}$, the number of levels.
- $n_i$, the total number of cases from category i

## Value

theta, float. The Freeman Theta value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Freeman, L. C. (1965). *Elementary applied statistics: For students in behavioral science*. Wiley.

Jacobson, P. E. (1972). Applying measures of association to nominal-ordinal data. *The Pacific Sociological Review, 15*(1), 41–60. doi:10.2307/1388286

---

## `es_glass_delta` *Glass Delta*

---

### Description

An effect size measure when comparing two means, with a specified control group.

### Usage

```
es_glass_delta(
  catField,
  scaleField,
  categories = NULL,
  dmu = 0,
  control = NULL
)
```

### Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scaleField` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `dmu` | Optional difference according to null hypothesis (default is 0) |
| `control` | Optional to indicate which category to use as control group. Default is first category found. |

### Details

The formula used is (Glass, 1976, p. 7):

$$\delta = \frac{\bar{x}_1 - \bar{x}_2}{s_2}$$

With:

$$s_2 = \sqrt{\frac{\sum_{i=1}^{n_2} \left(x_{2,i} - \bar{x}_2\right)^2}{n_2 - 1}}$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{i,j}}{n_i}$$

*Symbols used:*

- $x_{i,j}$ the j-th score in category i
- $n_i$ the number of scores in category i

Glass actually uses a ‘control group’ and $s_2$ is then the standard deviation of the control group.

### Value

Glass Delata value

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Glass, G. V. (1976). Primary, secondary, and meta-analysis of research. *Educational Researcher, 5*(10), 3–8. https://doi.org/10.3102/0013189X005010003

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['age']
ex1 = replace(ex1, ex1=="89 OR OLDER", "90")
es_glass_delta(df1['sex'], ex1)

#Example 2: vectors
scores = c(20,50,80,15,40,85,30,45,70,60, NA, 90,25,40,70,65, NA, 70,98,40)
groups = c("nat","int","int","nat","int","int","nat","nat","int","int",
"int","int","int","int","nat","int",NA,"nat","int","int")
es_glass_delta(groups, scores)
```

---

## es_goodman_kruskal_lambda

*Goodman-Kruskal Lambda*

### Description

Goodman-Kruskal Lambda

### Usage

```
es_goodman_kruskal_lambda(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  ties = "first"
)
```

### Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `categories1` | optional, categories to use for field1 |
| `categories2` | optional, categories to use for field2 |
| `ties` | c("first", "random", "average") optional to indicate what to do in case of multi-modal situations. |

**Details**

The formula used is (Goodman & Kruskal, 1954, p. 743):

$$\lambda_{Y|X} = \frac{\left(\sum_{i=1}^{r} f_{i,max}\right) - C_{max}}{n - C_{max}}$$

$$\lambda_{X|Y} = \frac{\left(\sum_{j=1}^{c} f_{max,j}\right) - R_{max}}{n - R_{max}}$$

$$\lambda = \frac{\left(\sum_{i=1}^{r} f_{i,max}\right) + \left(\sum_{j=1}^{c} f_{max,j}\right) - C_{max} - R_{max}}{2 \times n - C_{max} - R_{max}}$$

The asymptotic standard errors are calculated using (Hartwig, 1973, p. 308):

$$ASE\left(\lambda_{Y|X}\right)_0 = \sqrt{\left(\sum_{i,j}\left(F_{i,j} \times \left(\delta_{i,j}^{c} - \delta_{j}^{c}\right)^2\right)\right) - \frac{\left(\left(\sum_{i=1}^{r} f_{i,max}\right) - C_{max}\right)^2}{n}} n - C_{max}$$

$$ASE\left(\lambda_{X|Y}\right)_0 = \sqrt{\left(\sum_{i,j}\left(F_{i,j} \times \left(\delta_{i,j}^{r} - \delta_{i}^{r}\right)^2\right)\right) - \frac{\left(\left(\sum_{j=1}^{c} f_{max,j}\right) - R_{max}\right)^2}{n}} n - R_{max}$$

$$ASE\left(\lambda\right)_0 = \sqrt{\left(\sum_{i,j}\left(F_{i,j} \times \left(\delta_{i,j}^{c} - \delta_{j}^{c} + \delta_{i,j}^{r} - \delta_{i}^{r}\right)^2\right)\right) - \frac{\left(\left(\sum_{i=1}^{r} f_{i,max}\right) - C_{max} + \left(\sum_{j=1}^{c} f_{max,j}\right) - R_{max}\right)^2}{n}}$$

$$ASE\left(\lambda_{Y|X}\right)_1 = \sqrt{\frac{\left(n - \sum_{i=1}^{r} f_{i,max}\right) \times \left(\left(\sum_{i=1}^{r} f_{i,max}\right) + C_{max} - 2 \times \sum_{i,j}\left(F_{i,j} \times \delta_{i,j}^{c} \times \delta_{j}^{c}\right)\right)}{\left(n - C_{max}\right)^3}}$$

$$ASE\left(\lambda_{X|Y}\right)_1 = \sqrt{\frac{\left(n - \sum_{j=1}^{c} f_{max,j}\right) \times \left(\left(\sum_{j=1}^{c} f_{max,j}\right) + R_{max} - 2 \times \sum_{i,j}\left(F_{i,j} \times \delta_{i,j}^{r} \times \delta_{i}^{r}\right)\right)}{\left(n - R_{max}\right)^3}}$$

$$ASE\left(\lambda\right)_1 = \frac{\left(\sum_{i,j}\left(F_{i,j} \times \left(\delta_{i,j}^{c} + \delta_{i,j}^{r} - \delta_{j}^{c} - \delta_{i}^{r} + \lambda \times \left(\delta_{j}^{c} + \delta_{i}^{r}\right)\right)^2\right)\right) - 4 \times n \times \lambda^2}{2 \times n - C_{max} - R_{max}}$$

With:

$$\delta_{i,j}^{c} \begin{cases} 1 & \text{if } j = \text{column index} f_{i,max} \\ 0 & \text{else} \end{cases}$$

$$\delta_{j}^{c} \begin{cases} 1 & \text{if } j = \text{column index} C_{max} \\ 0 & \text{else} \end{cases}$$

$$\delta_{i,j}^{r} \begin{cases} 1 & \text{if } i = \text{row index} f_{max,j} \\ 0 & \text{else} \end{cases}$$

$$\delta_{i}^{r} \begin{cases} 1 & \text{if } i = \text{column index} R_{max} \\ 0 & \text{else} \end{cases}$$

The test is performed using:

$$z_i = \frac{\lambda_i}{ASE\left(i\right)_0}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_i\right|\right)\right)$$

*Symbols used*

- $F_{i,j}$ the absolute frequency (observed count) from row i and column j.
- $c$ the number of columns
- $r$ the number of rows
- $R_i$ row total of row i, it can be calculated using $R_i = \sum_{j=1}^{c} F_{i,j}$
- $C_j$ column total of column j, it can be calculated using $C_j = \sum_{i=1}^{r} F_{i,j}$
- $n$ the total number of cases, it can be calculated in various ways, $n = \sum_{j=1}^{c} C_j = \sum_{i=1}^{r} R_i = \sum_{i=1}^{r} \sum_{j=1}^{c} F_{i,j}$
- $f_{i,max}$ is the maximum count of row i, i.e. $f_{i,max} = \max\left\{F_{i,1}, F_{i,2}, \dots, F_{i,c}\right\}$
- $f_{max,j}$ is the maximum count of column j, i.e. $f_{max,j} = \max\left\{F_{1,j}, F_{2,j}, \dots, F_{r,j}\right\}$
- $R_{max}$ is the maximum of the row totals, i.e. $R_{max} = \max\left\{R_1, R_2, \dots, R_r\right\}$
- $C_{max}$ is the maximum of the column totals, i.e. $C_{max} = \max\left\{C_1, C_2, \dots, C_c\right\}$
- $\Phi\left(\dots\right)$ the cumulative density function of the standard normal distribution

Unfortunately not much is written about how to deal with situations if more than one row / column / cell has the highest (i.e. a multimodal situation). Hartwig proposed three options in case multimodal situation occurs: choose random, choose the largest ASE, or average them. This function can allow you to simply choose the first only (I think SPSS uses this), average them, or simply choose one at random.

## Value

dataframe with the effect size value, asymptotic standard error (both assuming null and alternative), the test statistic, and p-value

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Goodman, L. A., & Kruskal, W. H. (1954). Measures of Association for Cross Classifications. *Journal of the American Statistical Association, 49*(268), 732–764. https://doi.org/10.2307/2281536

Gray, L. N., & Campbell, R. (1975). Statistical significance of the Lambda coefficients: A comment. *Behavioral Science, 20*(4), 258–259. https://doi.org/10.1002/bs.3830200407

Hartwig, F. (1973). Statistical significance of the lambda coefficients. *Behavioral Science, 18*(4), 307–310. https://doi.org/10.1002/bs.3830180409

SPSS. (2006). SPSS 15.0 algorithms.

---

`es_goodman_kruskal_tau`

## *Goodman-Kruskal tau*

---

### Description

Goodman-Kruskal tau

### Usage

```
es_goodman_kruskal_tau(field1, field2, categories1 = NULL, categories2 = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `categories1` | optional, categories to use for field1 |
| `categories2` | optional, categories to use for field2 |

### Details

The formula used is (Goodman & Kruskal, 1954, p. 759):

$$\tau_{Y|X} = \frac{n\times\sum_{i=1}^{r}\sum_{j=1}^{c}\frac{F_{i,j}^2}{R_i} - \sum_{j=1}^{c}C_j^2}{n^2 - \sum_{j=1}^{c}C_j^2}$$

$$sig. = 1 - \chi^2\left(\left(n-1\right)\times\left(c-1\right)\times\tau_{Y|X}, df\right)$$

$$\tau_{X|Y} = \frac{n\times\sum_{i=1}^{r}\sum_{j=1}^{c}\frac{F_{i,j}^2}{C_i} - \sum_{i=1}^{r}R_i^2}{n^2 - \sum_{i=1}^{r}R_i^2}$$

$$sig. = 1 - \chi^2\left(\left(n-1\right)\times\left(r-1\right)\times\tau_{X|Y}, df\right)$$

With:

$$df = (r-1)\times(c-1)$$

*Symbols used*

- $F_{i,j}$ the absolute frequency (observed count) from row i and column j.
- $c$ the number of columns
- $r$ the number of rows
- $R_i$ row total of row i, it can be calculated using $R_i = \sum_{j=1}^{c} F_{i,j}$
- $C_j$ column total of column j, it can be calculated using $C_j = \sum_{i=1}^{r} F_{i,j}$
- $n$ the total number of cases, it can be calculated in various ways, $n = \sum_{j=1}^{c} C_j = \sum_{i=1}^{r} R_i = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$
- $\chi\left(\ldots,\ldots\right)$ the cumulative density function of the chi-square distribution

Light and Margolin developed a R2 measure for categorical data, they proposed a test CATANOVA (Categorical Anova) for this measure. This was a chi-square test (p. 538). Sarndal (1974, p. 178) concluded that R2 from Light and Mangolin, was the same as Goodman-Kendal tau, and uses their test for tau. Margolin and Light (1974) reach the same conclusion and proof the equivelance.

### Value

dataframe with the effect size value, the test statistic, degrees of freedom, and p-value

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

Goodman, L. A., & Kruskal, W. H. (1954). Measures of association for cross classifications. *Journal of the American Statistical Association, 49*(268), 732–764. https://doi.org/10.2307/2281536

Light, R. J., & Margolin, B. H. (1971). An analysis of variance for categorical data. *Journal of the American Statistical Association, 66*(335), 534–544. https://doi.org/10.1080/01621459.1971.10482297

Margolin, B. H., & Light, R. J. (1974). An analysis of variance for categorical data, II: Small sample comparisons with chi square and other competitors. *Journal of the American Statistical Association, 69*(347), 755–764. https://doi.org/10.1080/01621459.1974.10480201

Särndal, C. E. (1974). A comparative study of association measures. *Psychometrika, 39*(2), 165–187. https://doi.org/10.1007/BF02291467

---

## `es_hedges_g_is` *Hedges g / Cohen ds (independent samples)*

### Description

An effect size measure when comparing two means. A few different variations are available. See the details for more information on them.

### Usage

```
es_hedges_g_is(
  catField,
  scaleField,
  categories = NULL,
  dmu = 0,
  varWeighted = TRUE,
  corr = c(NULL, "exact", "hedges", "durlak", "xue")
)
```

### Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scaleField` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `dmu` | Optional difference according to null hypothesis (default is 0) |
| `varWeighted` | Optional boolean to indicate the use of weighted variances or not. Default is TRUE. |
| `corr` | approximation to use. Either NULL (default), 'exact', 'hedges', 'durlak', 'xue' |

**Details**

The formula used is (Hedges, 1981, p. 110):

$$g = \frac{\bar{x}_1 - \bar{x}_2}{s_p}$$

With:

$$s_p = \sqrt{\frac{SS_1^2 + SS_2^2}{n - 2}}$$

$$SS_i = \sum_{j=1}^{n_i} \left(x_{i,j} - \bar{x}_i\right)^2$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{i,j}}{n_i}$$

*Symbols used:*

- $x_{i,j}$ the j-th score in category i
- $n_i$ the number of scores in category i

This is also what Cohen refers to as $d_s$ (Cohen, 1988, p. 66).

This uses by default the formula as shown above for $s_p$. However, sometimes the unweighted version is used. If *varWeighted=FALSE* the following will be used instead:

$$s_p = \sqrt{\frac{s_1^2 + s_2^2}{2}}$$

Hedges proposes the following exact bias correction (Hedges, 1981, p. 111):

$$g_c = g \times \frac{\Gamma\left(m\right)}{\Gamma\left(m - \frac{1}{2}\right) \times \sqrt{m}}$$

With:

$$m = \frac{df}{2}$$

$$df = n_1 + n_2 - 2 = n - 2$$

*Symbols used:*

- $df$ the degrees of freedom
- $n$ the sample size (i.e. the number of scores)
- $\Gamma\left(\ldots\right)$ the gamma function

The formula used for the approximation for this correction from Hedges (1981, p. 114) (appr="hedges"):

$$g_c = g \times \left(1 - \frac{3}{4 \times df - 1}\right)$$

This approximation can also be found in Hedges and Olkin (1985, p. 81) and Cohen (1988, p. 66)

The formula used for the approximation from Durlak (2009, p. 927) (appr="durlak"):

$$g_c = g \times \frac{n - 3}{n - 2.25} \times \sqrt{\frac{n - 2}{n}}$$

The formula used for the approximation from Xue (2020, p. 3) (appr="xue"):

$$g_c = g \times \sqrt[12]{1 - \frac{9}{df} + \frac{69}{2 \times df^2} - \frac{72}{df^3} + \frac{687}{8 \times df^4} - \frac{441}{8 \times df^5} + \frac{247}{16 \times df^6}}$$

## Value

A dataframe with:

| | |
|---|---|
| `g` | the effect size value |
| `version` | description of the effect size calculated |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Durlak, J. A. (2009). How to select, calculate, and interpret effect sizes. *Journal of Pediatric Psychology, 34*(9), 917–928. https://doi.org/10.1093/jpepsy/jsp004

Hedges, L. V. (1981). Distribution Theory for Glass's Estimator of Effect Size and Related Estimators. *Journal of Educational Statistics, 6*(2), 107–128. https://doi.org/10.2307/1164588

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. Academic Press.

Xue, X. (2020). Improved approximations of Hedges' g*. https://doi.org/10.48550/arXiv.2003.06675

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['age']
ex1 = replace(ex1, ex1=="89 OR OLDER", "90")
es_hedges_g_is(df1['sex'], ex1)

#Example 2: vectors
scores = c(20,50,80,15,40,85,30,45,70,60, NA, 90,25,40,70,65, NA, 70,98,40)
groups = c("nat","int","int","nat","int","int","nat","nat","int","int",
"int","int","int","int","nat","int",NA,"nat","int","int")
es_hedges_g_is(groups, scores)
```

---

# `es_hedges_g_os` *Hedges g (for one-sample)*

## Description

This function will calculate Hedges g (one-sample). An effect size measure that can be used with a test for a single mean (for example a one-sample Student t-test).

Hedges g is a correction for Cohen's d'. Actually Hedges (1981) didn't seem to have a one-sample version for Hedges g, and this correction is the one for Hedges g used for the independent samples.

TThe measure is also described at PeterStatistics.com

## Usage

```
es_hedges_g_os(data, mu = NULL, appr = NULL)
```

## Arguments

| | |
|---|---|
| `data` | vector or dataframe with the numeric scores |
| `mu` | optional parameter to set the hypothesized mean. If not used the midrange is used |
| `appr` | optional approximation to use, NULL will use exact. Either `NULL` (default), `"hedges"`, `"durlak"`, or `"xue"` |

## Details

The formula used for the exact method (appr=NULL) (Hedges, 1981, p. 111):

$$g = d' \times \frac{\Gamma\left(m\right)}{\Gamma\left(m - \frac{1}{2}\right) \times \sqrt{m}}$$

With:

$$m = \frac{df}{2}$$
$$df = n - 1$$

*Symbols used:*

- $d'$ Cohen's d for one-sample
- $df$ the degrees of freedom
- $n$ the sample size (i.e. the number of scores)
- $\Gamma\left(\dots\right)$ the gamma function

The formula used for the approximation from Hedges (1981, p. 114) (appr="hedges"):

$$g = d' \times \left(1 - \frac{3}{4 \times df - 1}\right)$$

The formula used for the approximation from Durlak (2009, p. 927) (appr="durlak"):

$$g = d' \times \frac{n - 3}{n - 2.25} \times \sqrt{\frac{n - 2}{n}}$$

The formula used for the approximation from Xue (2020, p. 3) (appr="xue"):

$$g = d' \times \sqrt[12]{1 - \frac{9}{df} + \frac{69}{2 \times df^2} - \frac{72}{df^3} + \frac{687}{8 \times df^4} - \frac{441}{8 \times df^5} + \frac{247}{16 \times df^6}}$$

Since Hedges g is a correction for Cohen d', it can be converted to a regular Cohen d and then rules of thumb for the interpertation could be used.

## Value

dataframe with

- *mu*, the hypothesized mean used, the effect size value, and method used
- *g*, Hedges g for a one-sample
- *version*, description of version used.

### Before, After and Alternatives

Before this you might want to perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z-Test.

After this you might want a rule-of-thumb for the effect size, first convert to regular Cohen d: `es_convert`, to convert Hedges g to Cohen d, use *fr = "cohendos"* and *to = "cohend"*. `th_cohen_d`, for rules-of-thumb for Cohen d.

Alternative Effect Sizes: `es_cohen_d_os`, for for Cohen d'. `es_common_language_os`, for the Common Language Effect Size.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Durlak, J. A. (2009). How to select, calculate, and interpret effect sizes. *Journal of Pediatric Psychology, 34*(9), 917–928. https://doi.org/10.1093/jpepsy/jsp004

Hedges, L. V. (1981). Distribution Theory for Glass's Estimator of Effect Size and Related Estimators. *Journal of Educational Statistics, 6*(2), 107–128. https://doi.org/10.2307/1164588

Xue, X. (2020). Improved approximations of Hedges' g*. https://doi.org/10.48550/arXiv.2003.06675

### Examples

```
#Example 1: Numeric dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2['Gen_Age']
es_hedges_g_os(ex1)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
es_hedges_g_os(ex2)
```

---

## `es_hedges_g_ps` *Hedges g (Paired Samples)*

### Description

Effect size measure for paired samples. This is very similar as Hedges g for independent samples.

### Usage

```
es_hedges_g_ps(
  field1,
  field2,
  dmu = 0,
  appr = c("none", "hedges", "durlak", "xue"),
  within = TRUE
)
```

## Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `dmu` | optional the difference according to null hypothesis (default is 0) |
| `appr` | approximation to use (see details), default is "none" |
| `within` | boolean to use a correction for correlated pairs |

## Details

The formula used is the same as for Cohen d_z

The same corrections can then be applied as for the independent samples version. See es_hedges_g_is() for details.

### Alternatives

*library(effsize)*

datF = na.omit(data.frame(field1, field2))

cohen.d(datF$field1, datF$field2, paired=TRUE, within=TRUE, hedges.correction=TRUE)

cohen.d(datF$field1, datF$field2, paired=TRUE, within=FALSE, hedges.correction=TRUE)

## Value

A dataframe with:

| | |
|---|---|
| `g` | the Hedges g value |
| `version` | version that was used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

# `es_hodges_lehmann_is` *Hodges-Lehmann Estimator (Independent Samples)*

## Description

The Hodges-Lehmann estimate, is the median of all the possible differences between two sets of data. The authors (Hodges & Lehmann, 1963) describe it as the location shift that is needed to align two distributions (with similar distributions) as much as possible (p. 599).

It is sometimes incorrectly described as the difference between the two medians, but that is incorrect. It is not uncommon to have a different Hodges-Lehmann estimate than simply taking the difference between the two medians.

This measure is sometimes mentioned as an effect size measure for a Mann-Whitney U / Wilcoxon Rank Sum test (van Geloven, 2018), however since it is a median of the possible differences, it is not standardized (i.e. it doesn't range between two fixed values, and depends therefor on the data).

## Usage

```
es_hodges_lehmann_is(catField, scores, categories = NULL, levels = NULL)
```

## Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scores` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `levels` | Optional list with the ordinal text values in order |

## Details

The formula for the Hodges-Lehmann estimator with two samples is (Hodges & Lehmann, 1963, p. 602):

$$HL = \text{median}\left(x_i - y_j | 1 \leq i \leq n_x, 1 \leq j \leq n_y\right)$$

*Symbols used:*

- $x_i$ the i-th score in category x
- $x_j$ the j-th score in category y
- $n_i$ the number of scores in category i

There might be a faster method to actually determine this. Algorithm 616 (Monahan, 1984), but couldn't translate the Fortran to R

## Value

HL, the Hodges-Lehmann Estimator

| | |
|---|---|
| `CLE cat. 1` | the effect size for the first category |
| `CLE cat. 2` | the effect size for the second category |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Hodges, J. L., & Lehmann, E. L. (1963). Estimates of location based on rank tests. *The Annals of Mathematical Statistics, 34*(2), 598–611. doi:10.1214/aoms/1177704172

Monahan, J. F. (1984). Algorithm 616: Fast computation of the Hodges-Lehmann location estimator. *ACM Transactions on Mathematical Software, 10*(3), 265–270. doi:10.1145/1271.319414

van Geloven, N. (2018, March 13). Mann-Whitney U toets [Wiki]. Wikistatistiek. https://wikistatistiek.amc.nl/Mann-Whitney_U_toets

---

`es_jbm_e` *Johnston-Berry-Mielke E*

---

## Description

An effect size measure that could be used with a chi-square test or g-test.

The function is shown in this `YouTubevideo` and the test is also described at PeterStatistics.com

## Usage

```
es_jbm_e(chi2, n, minExp, test = c("chi", "g"))
```

## Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |
| `minExp` | the minimum expected count |
| `test` | optional to indicate if a chi-square tests, or a g (likelihood ratio) test was used. Either `"chi"` (default), or `"g"`. |

## Details

Two versions of this effect size. The formula for a chi-square test is:

$$E_{\chi^2} = \frac{q}{1-q} \times \left(\sum_{i=1}^{k}\frac{p_i^2}{q_i} - 1\right) = \frac{\chi_{GoF}^2 \times E_{min}}{n \times (n - E_{min})}$$

For a Likelihood Ratio (G) test:

$$E_L = -\frac{1}{\ln(q)} \times \sum_{i=1}^{k}\left(p_i \times \ln\left(\frac{p_i}{q_i}\right)\right) = -\frac{1}{\ln(q) \times \frac{\chi_L^2}{2 \times n}}$$

*Symbols used:*

- $q$ the minimum of all $q_i$
- $q_i$ the expected proportion in category i
- $p_i$ the observed proportion in category i
- $n$ the total sample size
- $E_min$ the minimum expected count
- $\chi_{GoF}^2$ the chi-square test statistic of a Pearson chi-square test of goodness-of-fit
- $\chi_L^2$ the chi-square test statistic of a likelihood ratio test of goodness-of-fit

Both formulas are from Johnston et al. (2006, p. 413)

A qualification rule-of-thumb could be obtained by converting this to Cohen's w

## Value

JBM's E value

### Before, After and Alternatives

Before this you will need a chi-square value. From either: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test. `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

After this you might want to use some rule-of-thumb for the interpretation by converting it to Cohen w: `es_convert` to convert JBM-E to Cohen w (using fr="jbme", to="cohenw", ex1=minExp/n). `th_cohen_w` for various rules-of-thumb for Cohen w.

Alternative effect sizes that use a chi-square value: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Johnston, J. E., Berry, K. J., & Mielke, P. W. (2006). Measures of effect size for chi-squared and likelihood-ratio goodness-of-fit tests. *Perceptual and Motor Skills, 103*(2), 412–414. https://doi.org/10.2466/pms.103.2.4
414

### Examples

```
chi2Value <- 3.106
n <- 19
minExp <- 3
es_jbm_e(chi2Value, n, minExp)
es_jbm_e(chi2Value, n, minExp, test="g")
```

---

## `es_jbm_r` *Berry-Johnston-Mielke R*

### Description

A chance-corrected version of eta-squared, as an effect size measure for a Cochran Q test.

### Usage

```
es_jbm_r(data, success = NULL)
```

### Arguments

| | |
|---|---|
| `data` | dataframe with the scores |
| `success` | indicator for what is considered a success (default is 1) |

## Details

The formula used (Berry et al., 2007 pp. 1237, 1239):

$$R = 1 - \frac{\delta}{\mu_\delta}$$

With:

$$\mu_\delta = \frac{2}{n \times (n-1)} \times \left(\sum_{i=1}^{n} p_i\right) \times \left(n - \sum_{i=1}^{n} p_i\right) - \sum_{i=1}^{n} p_i \times (1 - p_i)$$

$$\delta = \frac{1}{k \times \binom{n}{k}} \times \sum_{c=1}^{k} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} |x_{i,c} - x_{j,c}|$$

$$p_i = \frac{\sum_{j=1}^{k}}{k}$$

*Symbols used*

- $n$ the number of rows
- $k$ the number of columns
- $x_{i,j}$ the score in row i and column j

The function actually uses for:

$$\sum_{c=1}^{k} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} |x_{i,c} - x_{j,c}| = \sum_{j=1}^{k} C_j \times \left(n - \sum_{j=1}^{k} C_j\right)$$

With:

$$C_j = \sum_{i=1}^{n} x_{i,j}$$

The original article has in the equation for $\mu_\delta$ the first factor written as $\frac{2}{k \times (k-1)}$. In personal communication with one of the authors Alexis (2014) indicated this was wrong and $n$ should be used.

## Value

| | |
|---|---|
| `R` | the effect size measure |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Alexis. (2014, September 7). Answer to “Effect size of Cochran’s Q.” Cross Validated. https://stats.stackexchange.com/a/

Berry, K. J., Johnston, J. E., & Mielke, P. W. (2007). An alternative measure of effect size for Cochran’s Q test for related proportions. *Perceptual and Motor Skills, 104*(3_suppl), 1236–1242. doi:10.2466/pms.104.4.1236-1242

---

# `es_kendall_w` *Eta Squared (Maximum Corrected) for Cochran Q*

---

## Description

Eta Squared (Maximum Corrected) for Cochran Q

## Usage

```
es_kendall_w(Q, n, k)
```

## Arguments

| | |
|---|---|
| `Q` | the Cochran Q statistic |
| `n` | the sample size (number of rows) |
| `k` | the number of variables (number of columns) |

## Details

The formula used (Serlin et al., 1982 p. 788):

$$\eta_Q^2 = \frac{Q}{n \times (k - 1)}$$

*Symbols used*

- $n$ the number of rows
- $k$ the number of columns
- $Q$ the Cochran Q statistic

The Cochran Q statistic can be obtained using *ts_cochran_q()* function. The number of rows and columns of a dataframe with R's *nrow(dataframe)* and *ncol(dataframe)* functions.

## Value

| | |
|---|---|
| `es` | the effect size measure |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Serlin, R. C., Carr, J., & Marascuilo, L. A. (1982). A measure of association for selected nonparametric procedures. *Psychological Bulletin, 92*(3), 786–790. https://doi.org/10.1037/0033-2909.92.3.786

---

`es_odds_ratio` *Odds Ratio*

---

## Description

Determines the odds ratio from a 2x2 table.

Odds can sometimes be reported as 'a one in five odds', but sometimes as 1 : 4. This later notation is less often seen, but means for every one event on the left side, there will be four on the right side.

The Odds is the ratio of that something will happen, over the probability that it will not. For the Odds Ratio, we compare the odds of the first category with the second group.

If the result is 1, it indicates that one variable has no influence on the other. A result higher than 1, indicates the odds are higher for the first category. A result lower than 1, indicates the odds are lower for the first.

## Usage

```
es_odds_ratio(field1, field2, categories1 = NULL, categories2 = NULL)
```

## Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with selection and/or order for categories of field1 |
| `categories2` | : optional list with selection and/or order for categories of field2 |

## Details

The formula used is (Fisher, 1935, p. 50):

$$OR = \frac{a/c}{b/d} = \frac{a \times d}{b \times c}$$

*Symbols used:*

- $a$ the count in the top-left cell
- $b$ the count in the top-right cell
- $c$ the count in the bottom-left cell
- $d$ the count in the bottom-right cell
- $\Phi\left(\dots\right)$ the cumulative density function of the standard normal distribution

As for the test (McHugh, 2009, p. 123):

$$sig. = 2 \times \left(1 - \Phi\left(\left|z\right|\right)\right)$$

With:

$$SE = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$$

$$z = \frac{\ln\left(OR\right)}{SE}$$

The p-value is for the null-hypothesis that the population OR is 1.

The term Odds Ratio can for example be found in Cox (1958, p. 222).

### Value

Dataframe with:

| | |
|---|---|
| `OR` | the odds ratio |
| `n` | the sample size |
| `statistic` | the test statistic (z-value) |
| `p-value` | the significance (p-value) |

### Alternatives

R's *stats* library has a function that also shows an odds ratio: *fisher.test()*

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological), 20*(2), 215–232. https://doi.org/10.1111/j.2517-6161.1958.tb00292.x

Fisher, R. A. (1935). The logic of inductive inference. *Journal of the Royal Statistical Society, 98*(1), 39–82. https://doi.org/10.2307/2342435

McHugh, M. (2009). The odds ratio: Calculation, usage, and interpretation. *Biochemia Medica, 19*(2), 120–126. https://doi.org/10.11613/BM.2009.011

### See Also

`th_odds_ratio`, rules of thumb for odds ratio

`es_convert`, to convert an odds ratio to Yule Q, Yule Y, or Cohen d.

### Examples

```
#Example: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
es_odds_ratio(df1[['mar1']], df1[['sex']], categories1=c("WIDOWED", "DIVORCED"))
```

---

## `es_omega_sq` *Omega Squared*

### Description

An effect size measure for a one-way ANOVA. It indicates the the strength of the categories on the scale field. A 0 would indicate no influence, and 1 a perfect relationship.

Although a popular belief is that $\omega^2$ is preferred over $\epsilon^2$ (Keselman, 1975), a later study actually showed that $\epsilon^2$ might be preferred (Okada, 2013).

### Usage

```
es_omega_sq(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used (Kirk, 1996, p. 751):

$$\omega^2 = \frac{(F-1)\times df_b}{df_b\times(F-1)+n}$$

There are quite some variations on the formula above, all giving the same final result.

Hays (1973, p. 486) and Albers and Lakens (2018, p. 194):

$$\omega^2 = \frac{F-1}{\frac{df_w+1}{df_b}+F}$$

Caroll and Nordholm (1975, p. 547)

$$\omega^2 = \frac{F-1}{\frac{N-k+1}{k-1}+F}$$

Hays (1973, p. 485):

$$\omega^2 = \frac{SS_b-(k-1)\times MS_w}{SS_t+MS_w}$$

Olejnik and Algina (2003, p. 435):

$$\omega^2 = \frac{SS_b-df_b\times MS_w}{SS_b+(n-df_b)\times MS_w}$$

*Symbols*

- $x_{i,j}$ the i-th score in category j
- $n$ the total sample size
- $n_j$ the number of scores in category j
- $k$ the number of categories
- $\bar{x}_j$ the mean of the scores in category j
- $MS_i$ the mean square of i
- $SS_i$ the sum of squares of i (sum of squared deviation of the mean)
- $df_i$ the degrees of freedom of i
- $b$ is between = factor = treatment = model
- $w$ is within = error (the variability within the groups)
- $F_F$ the test statistic of the Fisher/Classic one-way ANOVA

The formula appears in many different formats. Hays (1973, p. 486) shows:

$$\hat{\omega}^2 = \frac{F_F - 1}{\frac{df_w + 1}{df_b} + F_F}$$

Which can also be found in Albers and Lakens (2018, p. 194).

Kirk (1996, p. 751) shows:

$$\hat{\omega}^2 = \frac{df_b \times (F_F - 1)}{df_b \times (F_F - 1) + n}$$

Instead of using the definitions for $df_b$ and $df_w$, Caroll and Nordholm (1975, p. 547) show the formula as:

$$\hat{\omega}^2 = \frac{F_F - 1}{\frac{n - k + 1}{k - 1} + F_F}$$

They also show Hays original formula (hays1) on p. 188.

Olejnik and Algina (2003, p. 435) use:

$$\hat{\omega}^2 = \frac{SS_b - df_b \times MS_w}{SS_b + (n - df_b) \times MS_w}$$

**Conversion**

To convert $\omega^2$ to $\epsilon^2$ use *es_convert(omegasq, from="omegasq", to="epsilonsq", ex1=MS_w, ex2=SS_t)*

**Alternatives**

*library(effectsize)*

anova_stats(aov(scores~groups))

omega_squared(aov(scores~groups))

## Value

the omega squared value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Albers, C., & Lakens, D. (2018). When power analyses based on pilot data are biased: Inaccurate effect size estimators and follow-up bias. *Journal of Experimental Social Psychology, 74*, 187–195. doi:10.1016/j.jesp.2017.09.004

Carroll, R. M., & Nordholm, L. A. (1975). Sampling characteristics of Kelley's $\epsilon$ and Hays' $\omega$. *Educational and Psychological Measurement, 35*(3), 541–554. doi:10.1177/001316447503500304

Hays, W. L. (1973). *Statistics for the social sciences* (2nd ed.). Holt, Rinehart and Winston.

Keselman, H. J. (1975). A Monte Carlo investigation of three estimates of treatment magnitude: Epsilon squared, eta squared, and omega squared. *Canadian Psychological Review / Psychologie Canadienne, 16*(1), 44–48. doi:10.1037/h0081789

Kirk, R. E. (1996). Practical significance: A concept whose time has come. *Educational and Psychological Measurement, 56*(5), 746–759. doi:10.1177/0013164496056005002

Okada, K. (2013). Is omega squared less biased? A comparison of three major effect size indices in one-way anova. *Behaviormetrika, 40*(2), 129–147. doi:10.2333/bhmk.40.129

Olejnik, S., & Algina, J. (2003). Generalized eta and omega squared statistics: Measures of effect size for some common research designs. *Psychological Methods, 8*(4), 434–447. doi:10.1037/1082-989X.8.4.434

---

## `es_pairwise_bin` *Binary Effect Size for Pairwise Test*

### Description

When using a pairwise post-hoc test for a single nominal variable, the pair has become binary. This function then can determine the effect size for each pair.

Options are to use Cohen g, Cohen h', or the Alternative Ratio.

### Usage

```
es_pairwise_bin(data, expCounts = NULL, es = "coheng")
```

### Arguments

| | |
|---|---|
| `data,` | list with the data |
| `expCounts` | dataframe, optional. The categories and expected counts |
| `es,` | string, optional. effect size to use.Either "coheng" (default), "cohenh", "ar" |

### Details

If expected counts are provided, for Cohen h' and the Alternative Ratio these expected counts are converted to expected proportions

See the separate functions of each effect size for more details.

- `es_cohen_g` for Cohen g
- `es_cohen_h_os` for Cohen h'
- `es_alt_ratio` for the Alternative Ratio

### Value

dataframe with

| | |
|---|---|
| `cat1` | label of first category in pair |
| `cat2` | label of second category in pair |
| `n1` | number of cases in first category |
| `n2` | number of cases in second category |
| `var` | followed by the effect size value |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### See Also

`ph_pairwise_bin`, performs a pairwise binomial test

---

## `es_pairwise_bin_ord` *Pairwise Binary-Ordinal Effect Sizes*

---

### Description

This function determines the effect size for each comparison in a post-hoc analysis of a nominal vs. ordinal variable (e.g. a Kruskal-Wallis test).

### Usage

```
es_pairwise_bin_ord(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  es = "cle"
)
```

### Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `es` | string, optional. the effect size to determine.Either "cle" (default), "rb" or "rosenthal" |

### Details

The function simply goes over each possible pair of categories from the *catField* (adjusted with *categories* if used). It then runs for only the scores of those two categories the Common Language Effect Size (Vargha-Delaney A) or (Glass) Rank Biserial (Cliff delta). If the Rosenthal correlation is requested, it will perform the post-hoc Dunn test to obtain the z-statistic.

### Value

dataframe with

| | |
|---|---|
| `cat1` | label of first category in pair |
| `cat2` | label of second category in pair |
| `effect size` | the value of the effect size |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### See Also

`es_common_language_is`, Common Language Effect size `r_rank_biserial_is`, rank biserial for independent samples `ph_dunn`, post-hoc Dunn test, used to obtain z-value for Rosenthal correlation

---

## `es_phi` *Pearson/Yule Phi Coefficient / Cole C2 / Mean Square Contingency*

---

### Description

After performing chi-square test the question of the effect size comes up. An obvious candidate to use in a measure of effect size is the test statistic, the $\chi^2$. One of the earliest and often mentioned measure uses this: the phi coefficient (or mean square contingency). Both Yule (1912, p. 596) and Pearson (1900, p. 12) mention this measure, and Cole (1949, p. 415) refers to it as Cole C2. It is also the same as Cohen's w (Cohen, 1988, p. 216), but Cohen does not specify it to be only for 2x2 tables.

It is interesting that this gives the same result, as if you would assign a 0 and 1 to each of the two variables categories, and calculate the regular correlation coefficient.

Pearson (1904, p. 6) calls the squared value (i.e. not taking the square root) the Mean Square Contingency.

### Usage

```
es_phi(field1, field2, categories1 = NULL, categories2 = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with order for categories of field1 |
| `categories2` | : optional list with order for categories of field2 |

### Details

The formula used is (Pearson, 1900, p. 12):

$$\phi = \frac{a \times d - b \times c}{\sqrt{R_1 \times R_2 \times C_1 \times C_2}}$$

*Symbols used:*

- $a$ the count in the top-left cell of the cross table
- $b$ the count in the top-right cell of the cross table
- $c$ the count in the bottom-left cell of the cross table
- $d$ the count in the bottom-right cell of the cross table
- $R_i$ the sum of counts in the i-th row
- $C_i$ the sum of counts in the i-th column

The formula is also sometimes expressed with a $\chi^2$ value (Pearson, 1904, p.6; Cohen, 1988, p. 216):

$$\phi = \sqrt{\frac{\chi^2}{n}}$$

Note that Cohen w did not limit the size of the table, but uses the same formula.

### Value

phi coefficient

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Cole, L. C. (1949). The measurement of interspecific associaton. *Ecology, 30*(4), 411–424. https://doi.org/10.2307/19324

Pearson, K. (1900). Mathematical Contributions to the Theory of Evolution. VII. On the Correlation of Characters not Quantitatively Measurable. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 195, 1–405.

Pearson, K. (1904). *Contributions to the Mathematical Theory of Evolution. XIII. On the theory of contingency and its relation to association and normal correlation*. Dulau and Co.

Yule, G. U. (1912). On the methods of measuring association between two attributes. *Journal of the Royal Statistical Society, 75*(6), 579–652. https://doi.org/10.2307/2340126

### See Also

`th_cohen_w`, rules of thumb for Cohen w

### Examples

```
#Example: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
es_phi(df1[['mar1']], df1[['sex']], categories1=c("WIDOWED", "DIVORCED"))
```

---

## `es_post_hoc_gof` *Effect Sizes for a Goodness-of-Fit Post-Hoc Analysis*

### Description

Determines an effect size for each test (row) from the results of ph_pairwise_bin(), ph_pairwise_gof(), ph_residual_bin(), or ph_residual_gof().

### Usage

```
es_post_hoc_gof(post_hoc_results, es = "auto", bergsma = FALSE)
```

### Arguments

| | |
|---|---|
| `post_hoc_results` | dataframe with the result of either ph_pairwise_bin(), ph_pairwise_gof(), ph_residual_bin(), or ph_residual_gof() |
| `es` | string optional, the effect size to determine. Either 'auto', 'coheng', 'cohenh', 'ar', 'cramerv', 'cohenw', 'jbme', 'fei', 'rosenthal' |
| `bergsma` | optional boolean. Use of Bergsma correction, only for Cramér V |

## Details

'auto' will use Cohen h for exact tests, Rosenthal correlation for z-tests and Cramér's V otherwise.

Cohen g ('coheng'), Cohen h ('cohenh') and Alternative Ratio ('ar') can all be used for any test.

Cramér V ('cramerv'), Cohen w ('cohenw'), Johnston-Berry-Mielke E ('jbme'), and Fei ('fei') can be used with chi-square tests (or likelihood ratio tests)

The Rosenthal Correlation ('rosenthal') can be used with a z-test (proportion/Wald/score/residual).

See the separate functions for each of these for details on the calculations.

## Value

a dataframe with for residual post-hoc:

**category** , the label of the category

**name effect size** , the effect size value

for pairwise post-hoc

**category 1** , the label of the first category

**category 2** , the label of the second category

**name effect size** , the effect size value

## Before, After and Alternatives

Before this a post-hoc test might be helpful: `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_pairwise_bin`, for Pairwise Binary Test. `ph_residual_gof_gof`, for Residuals Tests using Binary tests. `ph_residual_gof_bin`, for Residuals Using Goodness-of-Fit Tests.

After this you might want to use a rule-of-thumb for the interpretation: `th_post_hoc_gof`, for various rules-of-thumb.

Effect size in this function: `es_cohen_g`, for Cohen g. `es_cohen_h_os`, for Cohen h'. `es_alt_ratio`, for Alternative Ratio. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_cohen_w`, for Cohen's w. `es_jbm_e`, for Johnston-Berry-Mielke E. `es_fei`, for Fei. `r_rosenthal`, for Rosenthal Correlation if a z-value is available.

note: the effect size functions are not used themselves in this function, but the same formulas are used.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
# Get data
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
gssDf <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = gssDf['mar1']

# Perform a post-hoc test
post_hoc_test = ph_pairwise_bin(ex1, test='binomial')

# Determine the effect sizes
es_post_hoc_gof(post_hoc_test, es='cohenh')
```

---

## `es_rmsse` *Root Mean Square Standardized Effect Size (RMSSE)*

---

### Description

An effect size measure for a one-way ANOVA.

Similar as Hedges g, but for a one-way ANOVA. According to Wikipedia "this essentially presents the omnibus difference of the entire model adjusted by the root mean square" (2023).

### Usage

```
es_rmsse(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

### Details

The formula used (Steiger & Fouladi, 1997, pp. 244-245):

$$RMSSE = \sqrt{\frac{\delta}{(k-1)\times n}} = \sqrt{\frac{\sum_{i=1}^{k}\alpha_i^2}{(k-1)\times\sigma^2}}$$

With:

$$\delta = n \times \sum_{i=1}^{k}\left(\frac{\alpha_i}{\sigma}\right)^2$$

$$\alpha_i = \mu_i - \mu \approx \bar{x}_i - \bar{x}$$

$$\sigma \approx \sqrt{MS_w}$$

$$MS_w = \frac{SS_w}{df_w}$$

$$df_w = n - k$$

$$SS_w = SS_t - SS_b$$

$$SS_t = \sum_{j=1}^{k}\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}\right)^2$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

$$\bar{x} = \frac{\sum_{j=1}^{k} n_j \times \bar{x}_j}{n} = \frac{\sum_{j=1}^{k}\sum_{i=1}^{n_j} x_{i,j}}{n}$$

$$n = \sum_{j=1}^{k} n_j$$

*Symbols*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n$ the total sample size
- $n_j$ the number of scores in category j
- $\bar{x}_j$ the mean of the scores in category j
- $\bar{x} theoverallmean$
- $SS_w$ the within sum of squares (sum of squared deviation of the mean)
- $df_w$ the within degrees of freedom

Note that the original article refers to $\sigma^2$ as the error variance of the noncentral F-distribution. This can be approximated with $MS_w$ (Smith & Down, 2014, p. 2).

Zhang and Algina (2011) create a robust version of the RMSSE for one-way fixed effects anova.

### Value

the rmsse value

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Smith, B., & Dowd, M. (2014). One-way analysis of variance (ANOVA). Dalhousie University. https://www.mathstat.dal.ca/~stat2080/Fall14/Lecturenotes/anova1.pdf

Steiger, J. H., & Fouladi, R. T. (1997). *Noncentrality interval estimation and the evaluation of statistical models*. In L. L. Harlow, S. A. Mulaik, & J. H. Steiger, What if there were no significance tests? (pp. 221–257). Lawrence Erlbaum Associates.

Wikipedia. (2023). Effect size. In Wikipedia. https://en.wikipedia.org/w/index.php?title=Effect_size&oldid=117594862

Zhang, G., & Algina, J. (2011). A robust root mean square standardized effect size in one-way fixed-effects ANOVA. *Journal of Modern Applied Statistical Methods, 10*(1), 77–96. https://doi.org/10.22237/jmasm/13042222

---

## `es_scott_pi` *Scott Pi*

### Description

An effect size meaure, that measures the how strongly two raters or variables, agree with each other. Full agreement would result in a pi of 1.

The measure is very similar to Cohen's kappa. The difference is with the calculation of the expected marginal proportions. Cohen's kappa uses a squared geometric mean, while Scott's pi uses squared arithmetic means.

Scott developed this in criticism on Bennett-Alpert-Goldstein's S (see es_bag_s()).

### Usage

```
es_scott_pi(field1, field2, categories = NULL)
```

## Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |

## Details

The formula used (Scott, 1955, p. 323):

$$\pi = \frac{p_0 - p_e}{1 - p_e}$$

With:

$$P = \sum_{i=1}^{r} F_{i,i}$$

$$p_0 = \frac{P}{n}$$

$$p_e = \sum_{i=1}^{r}\left(\frac{R_i + C_i}{2 \times n}\right)^2$$

The asymptotic standard errors is calculated using (Scott, 1955, p. 325):

$$ASE = \sqrt{\left(\frac{1}{1 - p_e}\right)^2 \times \frac{p_0 \times (1 - p_0)}{n - 1}}$$

The p-value (significance) is then calculated using:

$$z_\pi = \frac{\pi}{ASE}$$

$$sig. = 2 \times (1 - \Phi(z_\kappa))$$

*Symbols used*

- $F_{i,j}$, the observed count in row i and column j.
- $r$, is the number of rows (categories in the first variable)
- $c$, is the number of columns (categories in the second variable)
- $n$, is the total number of scores
- $R_i$, the row total of row i.$R_i = \sum_{j=1}^{c} F_{i,j}$
- $C_j$, the column total of column j.$C_j = \sum_{i=1}^{r} F_{i,j}$

## Value

Dataframe with:

| | |
|---|---|
| `Scott pi` | the Scott pi value |
| `n` | the sample size |
| `statistic` | the test statistic (z-value) |
| `p-value` | the significance (p-value) |

## References

Scott, W. A. (1955). Reliability of content analysis: The case of nominal scale coding. *The Public Opinion Quarterly, 19*(3), 321–325.

---

`es_theil_u` *Theil U / Uncertainty Coefficient*

---

## Description

Theil U is a measure of nominal association. According to Wikipedia: "given Y, what fraction of the bits of X can we predict? In this case we can think of X as containing the total information, and of Y as allowing one to predict part of such information." (2022).

The term Theil U can also refer to two completely different measures, often used in forecasting and sometimes referred to as index of inequality.

## Usage

```
es_theil_u(field1, field2, categories1 = NULL, categories2 = NULL)
```

## Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `categories1` | optional, categories to use for field1 |
| `categories2` | optional, categories to use for field2 |

## Details

The formula used:

$$U_{Y|X} = \frac{H_X + H_Y - H_{XY}}{H_Y}$$

$$U_{X|Y} = \frac{H_X + H_Y - H_{XY}}{H_X}$$

$$U = 2 \times \frac{H_X + H_Y - H_{XY}}{H_X + H_Y}$$

With:

$$H_X = -\sum_{i=1}^{r} \frac{R_i}{n} \ln\left(\frac{R_i}{n}\right)$$

$$H_Y = -\sum_{j=1}^{c} \frac{C_j}{n} \ln\left(\frac{C_j}{n}\right)$$

$$H_X = -\sum_{i=1}^{r}\sum_{j=1}^{c} \frac{F_{i,j}}{n} \ln\left(\frac{F_{i,j}}{n}\right), \text{ for } F_{i,j} > 0$$

*Symbols used*

- $F_{i,j}$ the absolute frequency (observed count) from row i and column j.
- $c$ the number of columns
- $r$ the number of rows
- $R_i$ row total of row i, it can be calculated using $R_i = \sum_{j=1}^{c} F_{i,j}$
- $C_j$ column total of column j, it can be calculated using $C_j = \sum_{i=1}^{r} F_{i,j}$

- $n$ the total number of cases, it can be calculated in various ways, $n = \sum_{j=1}^{c} C_j = \sum_{i=1}^{r} R_i = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$

The asymptotic standard erros are calculated using:

$$ASE\left(U_{Y|X}\right)_1 = \frac{\sqrt{\sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(H_Y \times \ln\left(\frac{F_{i,j}}{R_i}\right) + \left(H_X - H_XY\right) \times \ln\left(\frac{C_j}{n}\right)\right)^2}}{n \times H_Y^2}$$

$$ASE\left(U_{X|Y}\right)_1 = \frac{\sqrt{\sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(H_X \times \ln\left(\frac{F_{i,j}}{C_j}\right) + \left(H_Y - H_XY\right) \times \ln\left(\frac{R_i}{n}\right)\right)^2}}{n \times H_X^2}$$

$$ASE\left(U\right)_1 = \frac{\sqrt{\sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(H_XY \times \ln\left(\frac{R_i \times C_j}{n^2}\right) - \left(H_X + H_Y\right) \times \ln\left(\frac{F_{i,j}}{n}\right)\right)^2}}{n \times \left(H_X + H_Y\right)^2}$$

$$ASE\left(U_{Y|X}\right)_0 = \frac{\sqrt{P - n \times \left(H_X + H_Y - H_{XY}\right)^2}}{n \times H_Y}$$

$$ASE\left(U_{X|Y}\right)_0 = \frac{\sqrt{P - n \times \left(H_X + H_Y - H_{XY}\right)^2}}{n \times H_X}$$

$$ASE\left(U\right)_0 = \frac{2 \times \sqrt{P - n \times \left(H_X + H_Y - H_{XY}\right)^2}}{n \times \left(H_X + H_Y\right)}$$

With:

$$P = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(\ln\left(\frac{R_i \times C_j}{n \times F_{i,j}}\right)\right)$$

The test statistic is:

$$T_i = \frac{U_i}{ASE\left(U_i\right)_0}$$

The formula's were taken from SPSS 15 Algorithms (2006, p. 117), unclear what the original source is, probably Theil (1970) or Theil (1972)

## Value

dataframe with

| | |
|---|---|
| `dependent` | the field used as dependent variable |
| `n` | the sample size |
| `value` | the Theil U value |
| `ASE_0` | the asymptotic standard error assuming the null hypothesis |
| `ASE_1` | the asymptotic standard error assuming the alternative hypothesis |
| `statistic` | the z-value |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

SPSS. (2006). SPSS 15.0 algorithms.

Theil, H. (1970). On the estimation of relationships involving qualitative variables. *American Journal of Sociology, 76*(1), 103–154. doi:10.1086/224909

Theil, H. (1972). *Statistical decomposition analysis: With applications in the social and administrative sciences* (Vol. 14). North-Holland Pub. Co.; American Elsevier Pub. Co.

Wikipedia. (2022). Uncertainty coefficient. In Wikipedia. https://en.wikipedia.org/w/index.php?title=Uncertainty_coeffi

---

## `es_tschuprow_t` *Tschuprow T*

### Description

Tschuprow T

### Usage

```
es_tschuprow_t(chi2, n, r, c, cc = NULL)
```

### Arguments

| | |
|---|---|
| `chi2` | the chi-square test statistic |
| `n` | the sample size |
| `r` | the number of categories in the first variable (i.e. the number of rows) |
| `c` | the number of categories in the second variable (i.e. the number of columns) |
| `cc` | c(NULL, "bergsma") optional to indicate correction to use (default is NULL) |

### Details

The formula used is:

$$T = \sqrt{\frac{\chi^2}{n \times \sqrt{(r-1) \times (c-1)}}}$$

*Symbols used*:

- $r$ the number of categories in the first variable (i.e. the number of rows)
- $c$ the number of categories in the second variable (i.e. the number of columns)
- $n$ the sample size, i.e. the sum of all frequencies
- $\chi^2$ the chi-square statistic

The formula is taken from Bergsma (2013, p. 324) who refers to Tschuprow (1925, 1939).

The Bergsma correction uses a different formula (Bergsma, 2013, pp. 324-325):

$$V_B = \sqrt{\frac{\tilde{\varphi}^2}{\sqrt{(\tilde{r}-1) \times (\tilde{c}-1)}}}$$

With:

$$\tilde{\varphi}^2 = \max\left(0, \varphi^2 - \frac{(r-1) \times (c-1)}{n-1}\right)$$

$$\tilde{r} = r - \frac{(r-1)^2}{n-1}$$

$$\tilde{c} = r - \frac{(c-1)^2}{n-1}$$

$$\varphi^2 = \frac{\chi^2}{n}$$

## Value

Tschuprow T value

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Bergsma, W. (2013). A bias-correction for Cramér's and Tschuprow's. *Journal of the Korean Statistical Society, 42*(3), 323–328. https://doi.org/10.1016/j.jkss.2012.10.002

Tschuprow, A. A. (1925). *Grundbegriffe und Grundprobleme der Korrelationstheorie*. B.G. Teubner.

Tschuprow, A. A. (1939). *Principles of the mathematical theory of correlation* (M. Kantorowitsch, Trans.). W. Hodge.

## Examples

```
chi2Val = 16.98975
n = 1941
nRows = 5
nCols = 2
es_tschuprow_t(chi2Val, n, nRows, nCols)
es_tschuprow_t(chi2Val, n, nRows, nCols, cc="bergsma")
```

---

`he_AS71` *Helper Function - Algorithm AS 71*

## Description

Helper Function - Algorithm AS 71

## Usage

```
he_AS71(S, N)
```

## Arguments

| | |
|---|---|
| `S` | the test statistic |
| `N` | the sample size |

## Details

Algorithm AS 71 (Best & Gipps, 1974) uses as test statistic:

$$S = \binom{n}{2} \times |\tau| = \frac{n \times (n-1)}{2} \times |\tau|$$

The Fortran code was translated to R by myself.

## Value

| | |
|---|---|
| `pValue` | upper tail p-value of Kendall tau Distribution |

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Best, D. J., & Gipps, P. G. (1974). Algorithm AS 71: The upper tail probabilities of Kendall's tau. *Applied Statistics, 23*(1), 98–100. https://doi.org/10.2307/2347062

---

# `he_AS89` *Helper Function - Algorithm AS 89*

## Description

Algorithm AS 89 (Best & Roberts, 1975) is for upper-tail probabilities

## Usage

```
he_AS89(n, IS)
```

## Arguments

| | |
|---|---|
| `n` | the number of scores (should be equal in both variables) |
| `IS` | the test statistic (see details) |

## Details

The test statistic $S$ defined as:

$$S = \sum_{i=1}^{n} d_i^2 = \sum_{i=1}^{n} \left(r_{x_i} - r_{y_i}\right)^2$$

Which if there are no ties is equal to:

$$S = \frac{\left(n^3 - n\right) \times \left(1 - r_s\right)}{6}$$

### Value

| | |
|---|---|
| `pValue` | the two-sided significance (p-value) |

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

Best, D. J., & Roberts, D. E. (1975). Algorithm AS 89: The upper tail probabilities of Spearman's rho. *Applied Statistics, 24*(3), 377–379. https://doi.org/10.2307/2347111

### Examples

```
n = 12
S = 8
he_AS89(n, S)
```

---

## `he_find_combinations` *Find Combinations*

---

### Description

Helper function for the multinomial cumulative distribution. Will return all possible combinations to distribute n items over k categories.

### Usage

```
he_find_combinations(n, k)
```

### Arguments

| | |
|---|---|
| `n` | int with the sample size |
| `k` | int with the number of categories |

### Value

A float with the requested probability

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

## `he_import_global` *Import global*

---

### Description

Only here to import the stats library

### Usage

```
he_import_global()
```

---

## `he_kendall` *Helper Function - Kendall Algorithm*

---

### Description

Helper Function - Kendall Algorithm

### Usage

```
he_kendall(n, c)
```

### Arguments

| | |
|---|---|
| `n` | the sample size |
| `c` | the number of concordant pairs |

### Details

An algorithm found at https://github.com/scipy/scipy/blob/v1.10.1/scipy/stats/_mstats_basic.py#L774-L898 was adapted. This refers to Kendall (1970), and uses the helper function *he_kendall(n, C)*. Where $C = n_c$, i.e. the number of concordant pairs. This algorithm already returns a two-tailed result.

### Value

| | |
|---|---|
| `pValue` | upper tail p-value of Kendall tau Distribution |

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## `he_owa_table` *One-Way ANOVA table*

### Description

Function to generate a one-way anova table

### Usage

```
he_owa_table(scores, groups)
```

### Arguments

| | |
|---|---|
| `scores` | a vector with the numeric scores |
| `groups` | a vector with the data containing the categories |

### Value

a dataframe with the ANOVA table

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### Examples

```
scores = c(20, 50, 80, 15, 40, 85, 30, 45, 70, 60, NA, 90, 25, 40, 70,
65, NA, 70, 98, 40, 65, 60, 35, NA, 50, 40, 75, NA, 65, 70, NA, 20, 80,
35, NA, 68, 70, 60, 70, NA, 80, 98, 10, 40, 63, 75, 80, 40, 90, 100, 33,
36, 65, 78, 50)
groups = c("R","H","D","R","H","D","R", "H", "D", "R", "H", "D", "R", "H",
"D", "R", "H", "D", "R", "H", "D", "R", "H", "D", "R", "H", "D", "R", "H",
"D", "R", "H", "D", "R", "H", "D", "R", "H", "D", "R", "H", "D", "R", "H",
"D", "R", "H", "D", "H", "D", "H", "H", "H", "H", "H")
he_owa_table(scores, groups)
```

## `he_permutations` *Helper Function for Permutations*

### Description

This function was posted by Museful (2013). It creates all possible permutations.

### Usage

```
he_permutations(n)
```

### Arguments

| | |
|---|---|
| `n` | the number of scores |

### Value

| | |
|---|---|
| `A` | all possible permutations of integers from 1 to n |

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

Museful. (2013, November 25). Answer to "Generating all distinct permutations of a list in R." Stack Overflow. https://stackoverflow.com/a/20199902/12149706

### Examples

```
he_permutations(5)
```

---

## `he_quantileIndexing` *Quartile Indexing*

### Description

Helper function for `me_quantiles` and `he_quantileIndexing` to return the index number of the quantiles.

### Usage

```
he_quantileIndexing(
  data,
  k = 4,
  method = c("sas1", "sas4", "hl", "excel", "hf8", "hf9")
)
```

### Arguments

| | |
|---|---|
| `data` | dataframe with scores as numbers |
| `k` | : number of quantiles |
| `method` | indexing method to use |

## Details

Six alternatives for the indexing is: Most basic (**SAS1**):

$$iQ_i = n \times p_i$$

**SAS4** method uses for indexing (SAS, 1990, p. 626; Snedecor, 1940, p. 43):

$$iQ_i = (n + 1) \times p_i$$

**Hog and Ledolter** use for their indexing (Hogg & Ledolter, 1992, p. 21; Hazen, 1914, p. ?):

$$iQ_i = n \times p_i + \frac{1}{2}$$

**MS Excel** uses for indexing (Gumbel, 1939, p. ?; Hyndman & Fan, 1996, p. 363):

$$iQ_i = (n - 1) \times p_i + 1$$

**Hyndman and Fan** use for their 8th version (Hyndman & Fan, 1996, p. 363):

$$iQ_i = \left(n + \frac{1}{3}\right) \times p_i + \frac{1}{3}$$

**Hyndman and Fan** use for their 9th version (Hyndman & Fan, 1996, p. 364):

$$iQ_i = \left(n + \frac{1}{4}\right) \times p_i + \frac{3}{8}$$

## Value

a vector with the quantiles

## Author(s)

P. Stikker. Companion Website, YouTube Channel

## References

Gumbel, E. J. (1939). La Probabilité des Hypothèses. Compes Rendus de l' Académie des Sciences, 209, 645–647.

Hazen, A. (1914). Storage to be provided in impounding municipal water supply. Transactions of the American Society of Civil Engineers, 77(1), 1539–1640. https://doi.org/10.1061/taceat.0002563

Hogg, R. V., & Ledolter, J. (1992). Applied statistics for engineers and physical scientists (2nd int.). Macmillan.

Hyndman, R. J., & Fan, Y. (1996). Sample quantiles in statistical packages. The American Statistician, 50(4), 361–365. https://doi.org/10.2307/2684934

SAS. (1990). SAS procedures guide: Version 6 (3rd ed.). SAS Institute.

Snedecor, G. W. (1940). Statistical methods applied to experiments in agriculture and biology (3rd ed.). The Iowa State College Press.

---

## `he_quantilesIndex` *Quantile Numeric Based on Index*

---

### Description

Helper function for `me_quartiles` to return the quartile as a number of the first and third quartile with different methods of rounding.

### Usage

```
he_quantilesIndex(
  data,
  k = 4,
  indexMethod = c("sas1", "sas4", "hl", "excel", "hf8", "hf9"),
 qLfrac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  qLint = c("int", "midpoint"),
 qHfrac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  qHint = c("int", "midpoint")
)
```

### Arguments

| | |
|---|---|
| `data` | dataframe with scores as numbers |
| `k` | : number of quantiles |
| `indexMethod` | optional to indicate which type of indexing to use |
| `qLfrac` | optional to indicate what type of rounding to use for quantiles below median |
| `qLint` | optional to indicate the use of the integer or the midpoint method for quantiles below median |
| `qHfrac` | optional to indicate what type of rounding to use for quantiles above median |
| `qHint` | optional to indicate the use of the integer or the midpoint method for quantiles above median |

### Details

If **the index is an integer** often that integer will be used to find the corresponding value in the sorted data. However, in some rare methods they argue to take the midpoint between the found index and the next one, i.e. to use:

$$iQ_i = iQ_i + \frac{1}{2}$$

If the index has a fractional part, we could use linear interpolation. It can be written as:

$$X\left[\lfloor iQ_i \rfloor\right] + \frac{iQ_i - \lfloor iQ_i \rfloor}{\lceil iQ_i \rceil - \lfloor iQ_i \rfloor} \times \left(X\left[\lceil iQ_i \rceil\right] - X\left[\lfloor iQ_i \rfloor\right]\right)$$

Where:

- $X\left[x\right]$ is the x-th score of the sorted scores
- $\lfloor \dots \rfloor$ is the function to always round down
- $\lceil \dots \rceil$ is the function to always round up

Or we can use 'rounding'. But there are different versions of rounding. Besides the already mentioned round down and round up versions:

- $\lfloor \dots \rceil$ to indicate rounding to the nearest even integer. A value of 2.5 gets rounded to 2, while 1.5 also gets rounded to 2. This is also referred to as *bankers* method.
- $[\dots]$ to indicate rounding to the nearest integer. A value that ends with .5 is then always rounded up.
- $\langle \dots \rangle$ to indicate to round a value ending with .5 always down

or even use the midpoint again i.e.:

$$\frac{\lfloor iQ_i \rfloor + \lceil iQ_i \rceil}{2}$$

## Value

A vector with the quantiles

## Author(s)

P. Stikker. Companion Website, YouTube Channel

---

## `he_quartileIndexing` *Quartile Indexing*

## Description

Helper function for `me_quartiles` and `he_quartileIndexing` to return the index number of the first and third quartile for different methods of determining this index.

## Usage

```
he_quartileIndexing(
  data,
 method = c("inclusive", "exclusive", "sas1", "sas4", "hl", "excel", "hf8", "hf9")
)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores as numbers |
| `method` | indexing method to use |

## Details

The **inclusive** method divides the data into two, and then includes the median in each half (if the sample size is odd). The first and third quarter are then the median of each of these two halves (Tukey, 1977, p. 32).

For the **inclusive** method, the index of the first quartile can be found using:

$$iQ_1 = \begin{cases} \frac{n+2}{4} & \text{if } n \bmod 2 = 0 \\ \frac{n+3}{4} & \text{else} \end{cases}$$

And the third quartile:

$$iQ_3 = \begin{cases} \frac{3\times n+2}{4} & \text{if } n \bmod 2 = 0 \\ \frac{3\times n+1}{4} & \text{else} \end{cases}$$

The **exclusive** method does the same as the inclusive method, but excludes the median in each half (if the sample size is odd) (Moore & McCabe, 1989, p. 33; Joarder & Firozzaman, 2001, p. 88).

For the **exclusive** method, the index of the first quartile can be found using:

$$iQ_1 = \begin{cases} \frac{n+2}{4} & \text{if } n \bmod 2 = 0 \\ \frac{n+1}{4} & \text{else} \end{cases}$$

And the third quartile:

$$iQ_3 = \begin{cases} \frac{3\times n+2}{4} & \text{if } n \bmod 2 = 0 \\ \frac{3\times n+3}{4} & \text{else} \end{cases}$$

Other methods use a different indexing. Six alternatives for the indexing is: Most basic (**SAS1**):

$$iQ_i = n \times p_i$$

**SAS4** method uses for indexing (SAS, 1990, p. 626; Snedecor, 1940, p. 43):

$$iQ_i = (n+1) \times p_i$$

**Hog and Ledolter** use for their indexing (Hogg & Ledolter, 1992, p. 21; Hazen, 1914, p. ?):

$$iQ_i = n \times p_i + \frac{1}{2}$$

**MS Excel** uses for indexing (Gumbel, 1939, p. ?; Hyndman & Fan, 1996, p. 363):

$$iQ_i = (n-1) \times p_i + 1$$

**Hyndman and Fan** use for their 8th version (Hyndman & Fan, 1996, p. 363):

$$iQ_i = \left(n + \frac{1}{3}\right) \times p_i + \frac{1}{3}$$

**Hyndman and Fan** use for their 9th version (Hyndman & Fan, 1996, p. 364):

$$iQ_i = \left(n + \frac{1}{4}\right) \times p_i + \frac{3}{8}$$

## Value

A dataframe with:

| | |
|---|---|
| `q1Index` | the index of the first (lower) quartile |
| `q3Index` | the index of the third (upper/higher) quartile |

## Author(s)

P. Stikker. Companion Website, YouTube Channel

## References

Gumbel, E. J. (1939). La Probabilité des Hypothèses. Compes Rendus de l' Académie des Sciences, 209, 645–647.

Hazen, A. (1914). Storage to be provided in impounding municipal water supply. Transactions of the American Society of Civil Engineers, 77(1), 1539–1640. https://doi.org/10.1061/taceat.0002563

Hogg, R. V., & Ledolter, J. (1992). Applied statistics for engineers and physical scientists (2nd int.). Macmillan.

Hyndman, R. J., & Fan, Y. (1996). Sample quantiles in statistical packages. The American Statistician, 50(4), 361–365. https://doi.org/10.2307/2684934

Joarder, A. H., & Firozzaman, M. (2001). Quartiles for discrete data. Teaching Statistics, 23(3), 86–89. https://doi.org/10.1111/1467-9639.00063

Moore, D. S., & McCabe, G. P. (1989). Introduction to the practice of statistics. W.H. Freeman.

SAS. (1990). SAS procedures guide: Version 6 (3rd ed.). SAS Institute.

Snedecor, G. W. (1940). Statistical methods applied to experiments in agriculture and biology (3rd ed.). The Iowa State College Press.

Tukey, J. W. (1977). Exploratory data analysis. Addison-Wesley Pub. Co.

---

# `he_quartilesIndex` *Quartile Numeric Based on Index*

---

## Description

Helper function for `me_quartiles` to return the quartile as a number of the first and third quartile with different methods of rounding.

## Usage

```
he_quartilesIndex(
  data,
  indexMethod = c("inclusive", "exclusive", "sas1", "sas4", "hl", "excel", "hf8",
    "hf9"),
  q1Frac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  q1Int = c("int", "midpoint"),
  q3Frac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  q3Int = c("int", "midpoint")
)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores as numbers |
| `indexMethod` | optional to indicate which type of indexing to use |
| `q1Frac` | optional to indicate what type of rounding to use for first quarter |
| `q1Int` | optional to indicate the use of the integer or the midpoint method for first quarter |
| `q3Frac` | optional to indicate what type of rounding to use for third quarter |
| `q3Int` | optional to indicate the use of the integer or the midpoint method for third quarter |

## Details

If **the index is an integer** often that integer will be used to find the corresponding value in the sorted data. However, in some rare methods they argue to take the midpoint between the found index and the next one, i.e. to use:

$$iQ_i = iQ_i + \frac{1}{2}$$

If the index has a fractional part, we could use linear interpolation. It can be written as:

$$X\left[\lfloor iQ_i \rfloor\right] + \frac{iQ_i - \lfloor iQ_i \rfloor}{\lceil iQ_i \rceil - \lfloor iQ_i \rfloor} \times \left(X\left[\lceil iQ_i \rceil\right] - X\left[\lfloor iQ_i \rfloor\right]\right)$$

Where:

- $X[x]$ is the x-th score of the sorted scores
- $\lfloor \dots \rfloor$ is the function to always round down
- $\lceil \dots \rceil$ is the function to always round up

Or we can use 'rounding'. But there are different versions of rounding. Besides the already mentioned round down and round up versions:

- $\lfloor \dots \rceil$ to indicate rounding to the nearest even integer. A value of 2.5 gets rounded to 2, while 1.5 also gets rounded to 2. This is also referred to as *bankers* method.
- $[\dots]$ to indicate rounding to the nearest integer. A value that ends with .5 is then always rounded up.
- $\langle \dots \rangle$ to indicate to round a value ending with .5 always down

or even use the midpoint again i.e.:

$$\frac{\lfloor iQ_i \rfloor + \lceil iQ_i \rceil}{2}$$

## Value

A dataframe with:

| | |
|---|---|
| `q1` | the first (lower) quartile |
| `q3` | the third (upper/higher) quartile |

## Author(s)

P. Stikker. Companion Website, YouTube Channel `he_spearman_permutation`

*Spearman Exact Distribution using Permutations*

### Description

This code in this function was posted by cuttlefish44 (2016) It performs a two-tailed exact test for Spearman rho.

### Usage

```
he_spearman_permutation(ord1, ord2)
```

### Arguments

| | |
|---|---|
| `ord1` | the numeric scores of the first variable |
| `ord2` | the numeric scores of the second variable |

### Details

The exact distribution is calculated using the following steps:

1. Determine all possible permutations of the scores in the first variable
2. Determine for each permutation the Spearman rho with the second variable
3. Count how often the Spearman rho is above the Spearman rho between the original two variables
4. Divide the results by $n!$

### Value

| | |
|---|---|
| `pValue` | the two-tailed p-value |

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

cuttlefish44. (2016, September 16). Answer to “Different methods for finding spearman’s coefficient produce diff p-values depending on presence of tied values.” Cross Validated. https://stats.stackexchange.com/a/235

### Examples

```
ord1 = c(5, 3, 3, 4, 3, 4, 3)
ord2 = c(5, 3, 3, 3, 3, 3, 5)

he_spearman_permutation(ord1, ord2)
```

## `he_tau_permutation` *Helper Function - Kendall Tau Permutation Test*

### Description

Helper Function - Kendall Tau Permutation Test

### Usage

```
he_tau_permutation(ord1, ord2)
```

### Arguments

| | |
|---|---|
| `ord1` | the numeric scores of the first variable |
| `ord2` | the numeric scores of the second variable |

### Details

Uses a permutation test to calculate the probability. It is an adaption to the code that was posted by cuttlefish44 (2016) online.

The exact distribution is calculated using the following steps:

1. Determine all possible permutations of the scores in the first variable
2. Determine for each permutation the Kendall tau with the second variable
3. Count how often the Spearman rho is above the Kendall tau between the original two variables
4. Divide the results by $n!$

### Value

| | |
|---|---|
| `pValue` | upper tail p-value of Kendall tau Distribution |

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

cuttlefish44. (2016, September 16). Answer to “Different methods for finding spearman’s coefficient produce diff p-values depending on presence of tied values.” Cross Validated. https://stats.stackexchange.com/a/235

### Examples

```
ord1 = c(5, 8, 6, 3, 2, 9)
ord2 = c(2, 1, 4, 5, 8, 7)
he_tau_permutation(ord1, ord2)
```

---

`me_consensus` *Consensus*

---

## Description

The Consensus is a measure of agreement or dispersion for ordinal data. If there is no agreement the value is 0, and with full agreement 1.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

## Usage

```
me_consensus(data, levels = NULL)
```

## Arguments

| | |
|---|---|
| `data` | a vector with the data |
| `levels` | optional to indicate the categories in order if data is non-numeric |

## Details

The formula used (Tastle et al., 2005, p. 98):

$$\text{Cns}\left(X\right) = 1 + \sum_{i=1}^k p_i \log_2\left(1 - \frac{\left|i - \mu_X\right|}{d_X}\right)$$

With:

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

$$d_X = k - 1$$

$$p_i = \frac{F_i}{n}$$

*Symbols used:*

- $F_i$ the frequency (count) of the i-th category (after they have been sorted)
- $n$ the sample size
- $k$ the number of categories.

## Value

cns the consensus score

### Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want some other descriptive measures: `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_median`, for the Median. `me_quantiles`, for Quantiles. `me_quartiles`, for Quartiles / Hinges. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Tastle, W. J., & Wierman, M. J. (2007). Consensus and dissention: A measure of ordinal dispersion. *International Journal of Approximate Reasoning, 45*(3), 531–545. https://doi.org/10.1016/j.ijar.2006.06.024

### Examples

```
# Example 1: Dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = studentDf[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_consensus(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_consensus(ex2)

#Example 3: Text data with
ex3 = c("a", "b", "f", "d", "e", "c")
order = c("a", "b", "c", "d", "e", "f")
me_consensus(ex3, levels=order)
```

---

## `me_hodges_lehmann_os` *Hodges-Lehmann Estimate (One-Sample)*

### Description

The Hodges-Lehmann Estimate (Hodges & Lehmann, 1963) for a one-sample scenario, is the median of the Walsh averages. The Walsh averages (Walsh, 1949a, 1949b) are the average of each possible pair by taking one score and combining it with each of the other scores. Note that each is only counted once, so taking the second and fifth score is the same as taking the fifth and the second, so only one of these is used. It does also include self-pairs, e.g. the third score and third score.

It is in the one-sample case therefor a measure of central tendancy and sometimes referred to as the pseudo median.

The measure is also described at PeterStatistics.com

## Usage

```
me_hodges_lehmann_os(scores, levels = NULL)
```

## Arguments

| | |
|---|---|
| `scores` | list with scores as numbers, or if text also provide levels |
| `levels` | optional vector with levels in order |

## Details

The formula used (Hodges & Lehmann, 1963, p. 599):

$$HL = \text{median}\left(\frac{x_i + x_j}{2}|i \leq i \leq j \leq n\right)$$

## Value

HL : float, the Hodges-Lehmann Estimate

## Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want some other descriptive measures: `me_consensus`, for the Consensus. `me_median`, for the Median. `me_quantiles`, for Quantiles. `me_quartiles`, for Quartiles / Hinges. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Hodges, J. L., & Lehmann, E. L. (1963). Estimates of location based on rank tests. *The Annals of Mathematical Statistics, 34*(2), 598–611. doi:10.1214/aoms/1177704172

Monahan, J. F. (1984). Algorithm 616: Fast computation of the Hodges-Lehmann location estimator. *ACM Transactions on Mathematical Software, 10*(3), 265–270. doi:10.1145/1271.319414

Walsh, J. E. (1949a). Applications of some significance tests for the median which are valid under very general conditions. Journal of the American Statistical Association, 44(247), 342–355. doi:10.1080/01621459.1949.10483311

Walsh, J. E. (1949b). Some significance tests for the median which are valid under very general conditions. *The Annals of Mathematical Statistics, 20*(1), 64–81. doi:10.1214/aoms/1177730091

## Examples

```
# Example 1: Dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = studentDf[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_hodges_lehmann_os(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_hodges_lehmann_os(ex2)

#Example 3: Text data with
ex3 = c("a", "b", "f", "d", "e", "c")
order = c("a", "b", "c", "d", "e", "f")
me_hodges_lehmann_os(ex3, levels=order)
```

---

# `me_mean` *Mean*

---

## Description

Different types of means can be determined using this function.

The mean is a measure of central tendency, to indicate the center.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

## Usage

```
me_mean(
  data,
  levels = NULL,
  version = "arithmetic",
  trimProp = 0.1,
  trimFrac = "down"
)
```

## Arguments

| | |
|---|---|
| `data,` | vector or dataframe with scores as numbers |
| `levels` | : list, optional coding to use |
| `version,` | optional mean to calculate. Either `"arithmetic"` (default), `"winsorized"`, `"trimmed"`, `"windsor"`, `"truncated"`, `"olympic"`, `"geometric"`, `"harmonic"`, `"midrange"`, or `"decile"` |
| `trimProp,` | optional to indicate the total proportion to trim. Default at 0.1 i.e. 0.05 from each side. |
| `trimFrac,` | optional parameter to indicate what to do if trimmed amount is non-integer. Either `"down"` (default), `"prop"`, `"linear"` |

## Details

### Arithmetic Mean

One of the three Pythagorean means, and the mean most people would assume if you ask them to calculate the mean.

It is the fulcrum of the distribution (Weinberg & Schumaker, 1962, p.19). One reference can for example be found in Aristotle (384-322 BC) (1850, p. 43).

The formula:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

### Harmonic Mean

The second of the three Pythagorean means:

$$H = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$$

### Geometric Mean

The third of the three Pythagorean means:

$$G = e^{\frac{1}{n}\times\sum_{i=1}^{n}\ln(x_i)}$$

### Olympic Mean

Simply ignore the maximum and minimum (only once) (Louis et al., 2023, p. 117):

$$OM = \frac{\sum_{i=2}^{n-1} x_i}{n-2}$$

### Mid Range

The average of the maximum and minimum (Lovitt & Holtzclaw, 1931, p. 91):

$$MR = \frac{\min x + \max x}{2}$$

### Trimmed

With a trimmed (Windsor/Truncated) mean we trim a fixed amount of scores from each side (Tukey, 1962, p. 17). Let $p_t$ be the proportion to trim, we then need to trim $n_t = \frac{p_t \times n}{2}$ from each side.

If this $n_t$ is an integer there isn't a problem, but if it isn't we have options. The first option is to simply round down, i.e. $n_l = \lfloor n_t \rfloor$. The trimmed mean is then:

$$\bar{x}_t = \frac{\sum_{i=n_t+1}^{n-n_l+1} x_i}{n - 2\times n_l}$$

This is used if *trimFrac = "down"* is set.

We could also use linear interpolation based on the number of scores to trim. We missed out on: $f = n_t - n_l$ on each side. So the first and last value we do include should only count for $1 - f$ each. The trimmed mean will then be:

$$\bar{x}_t = \frac{(x_{n_t+1} + x_{n-n_l+1}) \times (1-f) + \sum_{i=n_l+2}^{n-n_l} x_i}{n - 2\times n_t}$$

This is used if *trimFrac = "prop"* is set.

Alternative, we could take the proportion itself and use linear interpolation on that. The found $n_l$ will be $p_1 = \frac{n_l \times 2}{n}$ of the the total sample size. While if we had rounded up, we had used $p_2 = \frac{(n_l+1)\times 2}{n}$ of the the total sample size. Using linear interpolation we then get:

$$\bar{x}_t = \frac{p_t - p_1}{p_2 - p_1} \times (\bar{x}_{th} - \bar{x}_{tl}) + \bar{x}_{tl}$$

Where $\bar{x}_{tl}$ is the trimmed mean if $p_1$ would be used as a trim proportion, and $\bar{x}_{th}$ is the trimmed mean if $p_2$ would be used.

This is used if *trimFrac = "linear"* is set.

### Winsorized Mean

Similar as with a trimmed mean, but now the data is not removed, but replaced by the value equal to the nearest value that is still included (Winsor as cited in Dixon, 1960, p. 385).

$$W = \frac{n_l \times (x_{n_l+1} + x_{n-n_l}) + \sum_{n_l+1}^{n-n_l} x_i}{n}$$

## Value

res, the value of the mean

## Before, After and Alternatives

Before this you might want to create a binned frequency table or a visualisation: `tab_frequency_bins`, to create a binned frequency table. `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

After this you might want some other descriptive measures: `me_mode_bin`, for Mode for Binned Data. `me_variation`, for different Measures of Quantitative Variation.

Or a perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Aristotle. (1850). *The nicomachean ethics of Aristotle* (R. W. Browne, Trans.). Henry G. Bohn.

Dixon, W. J. (1960). Simplified estimation from censored normal samples. *The Annals of Mathematical Statistics, 31*(2), 385–391. https://doi.org/10.1214/aoms/1177705900

Louis, P., Núñez, M., & Xefteris, D. (2023). Trimming extreme reports in preference aggregation. *Games and Economic Behavior, 137*, 116–151. https://doi.org/10.1016/j.geb.2022.11.003

Lovitt, W. V., & Holtzclaw, H. F. (1931). *Statistics*. Prentice Hall.

Tukey, J. W. (1962). The future of data analysis. *The Annals of Mathematical Statistics, 33*(1), 1–67. https://doi.org/10.1214/aoms/1177704711

Weinberg, G. H., & Schumaker, J. A. (1962). *Statistics An intuitive approach*. Wadsworth Publishing.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Numeric dataframe
ex1 = df2['Gen_Age']
me_mean(ex1)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_mean(ex2)

#Example 3: Ordinal Pandas Series
ex3 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_mean(ex3, levels=order)
```

---

## `me_median` *Median*

### Description

Function to determine the median of a set of data. The median can be defined as "the middle value in a distribution, below and above which lie values with equal total frequencies or probabilities" (Porkess, 1991, p. 134). This means that 50% of the respondents scored equal or higher to the median, and also 50% of the respondents scored lower or equal.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

### Usage

```
me_median(data, levels = NULL, tieBreaker = c("between", "low", "high"))
```

### Arguments

| | |
|---|---|
| `data` | vector with the data |
| `levels` | optional list to indicate what values represent |
| `tieBreaker` | optional which to return if median falls between two values. Either `"between"` (default), `"low"`, or `"high"` |

### Details

The formula that is used, assuming the data has been sorted, is:

$$\tilde{x} = \begin{cases} x_{MI} & \text{if } MI = \lfloor MI \rfloor \\ \frac{x_{MI-0.5}+x_{MI+0.5}}{2} & \text{if } MI \neq \lfloor MI \rfloor \end{cases}$$

With:

$$MI = \frac{n+1}{2}$$

*Symbols used:*

- $n$ the sample size
- $x_i$ the i-th score of X, assuming X has been sorted.
- $MI$ the index of the median
- $\tilde{x}$ the median

If the number of scores is an odd number, and the median falls between two categories. With the *tieBreaker* it can then be set to return the lower value (low), upper (high), or average (between).

Some old references to the median are Pacioli (1523) in Italian, Cournot (1843, p. 120) in French, and Galton (1881, p. 246) in English.

### Value

the median

### Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want some other descriptive measures: `me_consensus`, for the Consensus. `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_quantiles`, for Quantiles. `me_quartiles`, for Quartiles / Hinges. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cournot, A. A. (1843). *Exposition de la théorie des chances et des probabilités*. L. Hachette.

Galton, F. (1881). Report of the anthropometric committee. *Report of the British Association for the Advancement of Science, 51*, 225–272.

Pacioli, L. (1523). *Summa de arithmetica geometria proportioni: Et proportionalita*. Paganino de Paganini.

Porkess, R. (1991). *The HarperCollins dictionary of statistics*. HarperPerennial.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Text dataframe
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_median(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_median(ex2)
```

```
#Example 3: Text data with between median
ex3 = c("a", "b", "f", "d", "e", "c")
order = c("a", "b", "c", "d", "e", "f")
me_median(ex3, levels=order)
me_median(ex3, levels=order, tieBreaker="low")
me_median(ex3, levels=order, tieBreaker="high")

#Example 4: Numeric data with between median
ex4 = c(1, 2, 3, 4, 5, 6)
me_median(ex4)
me_median(ex4, tieBreaker="low")
me_median(ex4, tieBreaker="high")
```

---

## `me_mode` *Mode*

---

### Description

The mode is a measure of central tendency and defined as “the abscissa corresponding to the ordinate of maximum frequency” (Pearson, 1895, p. 345). A more modern definition would be “the most common value obtained in a set of observations” (Weisstein, 2002).

The word mode might even come from the French word ’mode’ which means fashion. Fashion is what most people wear, so the mode is the option most people chose.

If one category has the highest frequency this category will be the modal category and if two or more categories have the same highest frequency each of them will be the mode. If there is only one mode the set is sometimes called unimodal, if there are two it is called bimodal, with three trimodal, etc. For two or more, thse term multimodal can also be used.

An advantage of the mode over many other measures of central tendency (like the median and mean), is that it can be determined for already nominal data types.

A video on the mode is available here.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

### Usage

```
me_mode(data, allEq = c("none", "all"))
```

### Arguments

| | |
|---|---|
| `data` | vector with the scores to determine the mode from |
| `allEq` | optional indicator on what to do if maximum frequency is equal for more than one category. Either `"none"` (default), or `"all"` |

### Details

One small controversy exists if all categories have the same frequency. In this case none of them has a higher occurence than the others, so none of them would be the mode (see for example Spiegel & Stephens, 2008, p. 64, Larson & Farber, 2014, p. 69). This is used when *allEq="none"* and the default.

On a rare occasion someone might argue that if all categories have the same frequency, then all categories are part of the mode since they all have the highest frequency. This is used when *allEq="all"*.

## Value

A dataframe with:

| | |
|---|---|
| `mode` | the mode(s) |
| `mode freq.` | frequency of the mode |

## Before, After and Alternatives

Before this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table. `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might want some variation measure: `me_qv`, for Measures of Qualitative Variation.

or perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

If you are looking to determine the mode of binned data use: `me_mode_bin`, for the mode with binned data

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Larson, R., & Farber, E. (2014). *Elementary statistics: Picturing the world* (6th ed.). Pearson.

Pearson, K. (1895). Contributions to the mathematical theory of evolution. II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London. (A.), 186*, 343–414. https://doi.org/10.1098/rsta.1895.0010

Spiegel, M. R., & Stephens, L. J. (2008). *Schaum's outline of theory and problems of statistics* (4th ed.). McGraw-Hill.

Weisstein, E. W. (2002). *CRC concise encyclopedia of mathematics* (2nd ed.). Chapman & Hall/CRC.

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
me_mode(ex1)

#Example 2: a list
ex2 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED", "DIVORCED",
"DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED", "DIVORCED",
"NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
me_mode(ex2)

#Example 3: Multi-Mode
ex3a = c(1, 1, 2, 3, 3, 4, 5, 6, 6)
```

```
me_mode(ex3a)
ex3b = c("MARRIED", "DIVORCED", "MARRIED", "DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED")
me_mode(ex3b)

#Example 4: All Equal
ex4a = c(1, 1, 2, 2, 3, 3, 6, 6)
me_mode(ex4a)
ex4b = c(1, 1, 2, 2, 3, 3, 6, 6)
me_mode(ex4b, allEq="all")
```

---

## `me_mode_bin` *Mode for Binned Data*

---

### Description

The mode is a measure of central tendency and defined as “the abscissa corresponding to the ordinate of maximum frequency” (Pearson, 1895, p. 345). A more modern definition would be “the most common value obtained in a set of observations” (Weisstein, 2002).

For binned data the mode is the bin with the highest frequency density. This will have the same result as using the highest frequency if all bins are of equal size. A frequency density is the frequency divided by the bin size (Zedeck, 2014, pp. 144-145). Different methods exist to narrow this down to a single value. See the notes for more info on this.

The word mode might even come from the French word ’mode’ which means fashion. Fashion is what most people wear, so the mode is the option most people chose.

If one category has the highest frequency this category will be the modal category and if two or more categories have the same highest frequency each of them will be the mode. If there is only one mode the set is sometimes called unimodal, if there are two it is called bimodal, with three trimodal, etc. For two or more, thse term multimodal can also be used.

An advantage of the mode over many other measures of central tendency (like the median and mean), is that it can be determined for already nominal data types.

### Usage

```
me_mode_bin(
  data,
  nbins = "sturges",
  bins = NULL,
  incl_lower = TRUE,
  adjust = 1,
  allEq = "none",
  value = "none"
)
```

### Arguments

| | |
|---|---|
| `data` | list or dataframe |
| `nbins` | optional, either the number of bins to create, or a specific method from the *tab_nbins()* function. Default is "sturges" |
| `bins` | optional dataframe with lower and upper bounds |

| | |
|---|---|
| `incl_lower` | optional boolean, to include the lower bound, otherwise the upper bound is included. Default is True |
| `adjust` | optional value to add or subtract to guarantee all scores will fit in a bin |
| `allEq` | optional indicator on what to do if maximum frequency is equal for more than one category. Either "none" (default) or "all" |
| `value` | optional which value to show in the output. Either "none" (default), "midpoint", or "quadratic" |

## Details

The function will use the **tab_frequency_bins()** function with the given parameters *nbins*, *bins*, *incl_lower* and *adjust*. See details of that function for more info.

### Value to return

If *value="midpoint"* is used the modal bin(s) midpoints are shown, using:

$$MP_m = \frac{UB_m + LB_m}{2}$$

Where $UB_m$ is the upper bound of the modal bin, and $LB_m$ the lower bound.

If *value="quadratic"* is used a quadratic curve is made from the midpoint of the bin prior to the modal bin, to the midpoint of the bin after the modal bin. This is done using:

$$M = LB_m + \frac{d_1}{d_1 + d_2} \times \left(UB_m - LB_m\right)$$

With:

$$d_1 = FD_m - FD_{m-1}$$

$$d_2 = FD_m - FD_{m+1}$$

Where $FD_m$ is the frequency density of the modal category.

### Multimode

One small controversy exists if all categories have the same frequency. In this case none of them has a higher occurence than the others, so none of them would be the mode (see for example Spiegel & Stephens, 2008, p. 64, Larson & Farber, 2014, p. 69). This is used when *allEq="none"* and the default.

On a rare occasion someone might argue that if all categories have the same frequency, then all categories are part of the mode since they all have the highest frequency. This is used when *allEq="all"*.

The function can return the bins that are the modal bins, by setting *value="none"*.

## Value

A dataframe with

| | |
|---|---|
| `mode` | the mode(s) |
| `mode fd` | frequency density of the mode |

### Before, After and Alternatives

Before this you might want to create a binned frequency table or a visualisation: `tab_frequency_bins`, to create a binned frequency table. `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

After this you might want some other descriptive measures: `me_mean`, for different types of mean. `me_variation`, for different Measures of Quantitative Variation.

Or a perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Larson, R., & Farber, E. (2014). *Elementary statistics: Picturing the world* (6th ed.). Pearson.

Pearson, K. (1895). Contributions to the mathematical theory of evolution. II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London. (A.), 186*, 343–414. https://doi.org/10.1098/rsta.1895.0010

Spiegel, M. R., & Stephens, L. J. (2008). *Schaum's outline of theory and problems of statistics* (4th ed.). McGraw-Hill.

Weisstein, E. W. (2002). *CRC concise encyclopedia of mathematics* (2nd ed.). Chapman & Hall/CRC.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
# Example 1: Numeric dataframe
ex1 = studentDf['Gen_Age']
myBins = data.frame(c(0, 20, 25, 30), c(20, 25, 30, 120))
me_mode_bin(ex1, bins=myBins)

# Example 2: Numeric list unimodal
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5)
myBins = data.frame(c(0, 3, 5), c(3, 5, 6))
me_mode_bin(ex2, bins=myBins)

# Example 3: Numeric list bimodal and using midpoint
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 6, 6, 6)
myBins = data.frame(c(1, 3, 5), c(3, 5, 7))
me_mode_bin(ex2, bins=myBins, value='midpoint')
```

## `me_quantiles` *Quantiles*

### Description

Quantiles split the data into k sections, each containing n/k scores. They can be seen as a generalisation of various 'tiles'. For example 4-quantiles is the same as the quartiles, 5-quantiles the same as quintiles, 100-quantiles the same as percentiles, etc.

Quite a few different methods exist to determine these. See the notes for more information.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

### Usage

```
me_quantiles(
  data,
  levels = NULL,
  k = 4,
  method = "own",
  indexMethod = c("sas1", "sas4", "hl", "excel", "hf8", "hf9"),
 q1Frac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  q1Int = c("int", "midpoint"),
 q3Frac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  q3Int = c("int", "midpoint")
)
```

### Arguments

| | |
|---|---|
| `data` | : list or dataframe |
| `levels` | : list, optional coding to use |
| `k` | : number of quantiles |
| `method` | : string, optional which method to use to calculate quartiles |
| `indexMethod` | : "sas1", "sas4", "excel", "hl", "hf8", "hf9", optional to indicate which type of indexing to use. Default is "sas1" |
| `q1Frac` | : "linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint", optional to indicate what type of rounding to use for quantiles below 50 percent. Default is "linear" |
| `q1Int` | : "int", "midpoint", optional to indicate the use of the integer or the midpoint method for first quarter. Default is "int" |
| `q3Frac` | : "linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint", optional to indicate what type of rounding to use for quantiles equal or above 50 percent. Default is "linear" |
| `q3Int` | : "int", "midpoint", optional to indicate the use of the integer or the midpoint method for quantiles equal or above 50 percent. Default is "int"<br>method can be set to "own" and then provide the next parameters, or any of the methods listed in the notes. |

## Details

To determine the quartiles a specific indexing method can be used. See **he_quantileIndexing()** for details on the different methods to choose from.

Then based on the indexes either linear interpolation or different rounding methods (bankers, nearest, down, up, half-down) can be used, or the midpoint between the two values. If the index is an integer either the integer or the mid point is used.

See the **he_quantilesIndex()** for details on this.

Note that the rounding method can even vary per quantile, i.e. the one used for the ones below the median being different than the one those equal or above.

I've come across the following methods:

| method | indexing | q1 integer | q1 fractional | q3 integer | q3 fractional |
|---|---|---|---|---|---|
| sas1 | sas1 | use int | linear | use int | linear |
| sas2 | sas1 | use int | bankers | use int | bankers |
| sas3 | sas1 | use int | up | use int | up |
| sas5 | sas1 | midpoint | up | midpoint | up |
| hf3b | sas1 | use int | nearest | use int | halfdown |
| sas4 | sas4 | use int | linear | use int | linear |
| ms | sas4 | use int | nearest | use int | halfdown |
| lohninger | sas4 | use int | nearest | use int | nearest |
| hl2 | hl | use int | linear | use int | linear |
| hl1 | hl | use int | midpoint | use int | midpoint |
| excel | excel | use int | linear | use int | linear |
| pd2 | excel | use int | down | use int | down |
| pd3 | excel | use int | up | use int | up |
| pd4 | excel | use int | halfdown | use int | nearest |
| pd5 | excel | use int | midpoint | use int | midpoint |
| hf8 | hf8 | use int | linear | use int | linear |
| hf9 | hf9 | use int | linear | use int | linear |

The following values can be used for the *method* parameter:

1. sas1 = parzen = hf4 = interpolated_inverted_cdf = maple3 = r4. (Parzen, 1979, p. 108; SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 363)
2. sas2 = hf3 = r3. (SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 362)
3. sas3 = hf1 = inverted_cdf = maple1 = r1 (SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 362)
4. sas4 = hf6 = minitab = snedecor = weibull = maple5 = r6 (Hyndman & Fan, 1996, p. 363; Weibull, 1939, p. ?; Snedecor, 1940, p. 43; SAS, 1990, p. 626)
5. sas5 = hf2 = CDF = averaged_inverted_cdf = r2 (SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 362)
6. hf3b = closest_observation
7. ms (Mendenhall & Sincich, 1992, p. 35)
8. lohninger (Lohninger, n.d.)
9. hl1 (Hogg & Ledolter, 1992, p. 21)
10. hl2 = hf5 = Hazen = maple4 = r5 (Hogg & Ledolter, 1992, p. 21; Hazen, 1914, p. ?)
11. maple2

12. excel = hf7 = pd1 = linear = gumbel = maple6 = r7 (Hyndman & Fan, 1996, p. 363; Freund & Perles, 1987, p. 201; Gumbel, 1939, p. ?)
13. pd2 = lower
14. pd3 = higher
15. pd4 = nearest
16. pd5 = midpoint
17. hf8 = median_unbiased = maple7 = r8 (Hyndman & Fan, 1996, p. 363)
18. hf9 = normal_unbiased = maple8 = r9 (Hyndman & Fan, 1996, p. 363)

*hf* is short for Hyndman and Fan who wrote an article showcasing many different methods, *hl* is short for Hog and Ledolter, *ms* is short for Mendenhall and Sincich, *jf* is short for Joarder and Firozzaman. *sas* refers to the software package SAS, *maple* to Maple, *pd* to Python's pandas library, and *r* to R.

The names *linear*, *lower*, *higher*, *nearest* and *midpoint* are all used by pandas quantile function and numpy percentile function. Numpy also uses *inverted_cdf*, *averaged_inverted_cdf*, *closest_observation*, *interpolated_inverted_cdf*, *hazen*, *weibull*, *median_unbiased*, and *normal_unbiased*.

## Value

results : the quantiles, or if levels are used also additionally text versions

## Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want some other descriptive measures: `me_consensus`, for the Consensus. `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_median`, for the Median. `me_quartiles`, for Quartiles / Hinges. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel

## References

Freund, J. E., & Perles, B. M. (1987). A new look at quartiles of ungrouped data. *The American Statistician, 41*(3), 200–203. https://doi.org/10.1080/00031305.1987.10475479

Galton, F. (1881). Report of the anthropometric committee. *Report of the British Association for the Advancement of Science, 51*, 225–272.

Gumbel, E. J. (1939). La Probabilité des Hypothèses. *Compes Rendus de l' Académie des Sciences, 209*, 645–647.

Hazen, A. (1914). Storage to be provided in impounding municipal water supply. *Transactions of the American Society of Civil Engineers, 77*(1), 1539–1640. https://doi.org/10.1061/taceat.0002563

Hogg, R. V., & Ledolter, J. (1992). *Applied statistics for engineers and physical scientists* (2nd int.). Macmillan.

Hyndman, R. J., & Fan, Y. (1996). Sample quantiles in statistical packages. *The American Statistician, 50*(4), 361–365. https://doi.org/10.2307/2684934

Langford, E. (2006). Quartiles in elementary statistics. *Journal of Statistics Education, 14*(3), 1–17. https://doi.org/10.1080/10691898.2006.11910589

Lohninger, H. (n.d.). Quartile. Fundamentals of Statistics. Retrieved April 7, 2023, from http://www.statistics4u.com/fun

McAlister, D. (1879). The law of the geometric mean. *Proceedings of the Royal Society of London, 29*(196–199), 367–376. https://doi.org/10.1098/rspl.1879.0061

Mendenhall, W., & Sincich, T. (1992). *Statistics for engineering and the sciences* (3rd ed.). Dellen Publishing Company.

Parzen, E. (1979). Nonparametric statistical data modeling. *Journal of the American Statistical Association, 74*(365), 105–121. https://doi.org/10.1080/01621459.1979.10481621

SAS. (1990). SAS procedures guide: Version 6 (3rd ed.). SAS Institute.

Siegel, A. F., & Morgan, C. J. (1996). *Statistics and data analysis: An introduction* (2nd ed.). J. Wiley.

Snedecor, G. W. (1940). *Statistical methods applied to experiments in agriculture and biology* (3rd ed.). The Iowa State College Press.

Vining, G. G. (1998). *Statistical methods for engineers*. Duxbury Press.

Weibull, W. (1939).* The phenomenon of rupture in solids*. Ingeniörs Vetenskaps Akademien, 153, 1–55.

## Examples

```
# Example 1: Dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = studentDf[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_quantiles(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_quantiles(ex2)

#Example 3: Text data with
ex3 = c("a", "b", "f", "d", "e", "c")
order = c("a", "b", "c", "d", "e", "f")
me_quantiles(ex3, levels=order)
```

---

# `me_quartiles` *Quartiles / Hinges*

## Description

The quartiles are at quarters of the data (McAlister, 1879, p. 374; Galton, 1881, p. 245). The median is at 50 percent, and the quartiles at 25 and 75 percent. Note that there are five quartiles, the minimum value is the 0-quartile, at 25 percent the first (or lower) quartile, at 50 percent the median a.k.a. the second quartile, at 75 percent the third (or upper) quartile, and the maximum as the fourth quartile.

Tukey (1977) also introduced the term Hinges and sorted the values in a W shape, where the bottom parts of the W are then the hinges.

There are quite a few different methods to determine the quartiles. This function has 19 different ones. See the details for a description.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

## Usage

```
me_quartiles(
  data,
  levels = NULL,
  method = "own",
  indexMethod = c("inclusive", "exclusive", "sas1", "sas4", "hl", "excel", "hf8",
    "hf9"),
  q1Frac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  q1Int = c("int", "midpoint"),
  q3Frac = c("linear", "down", "up", "bankers", "nearest", "halfdown", "midpoint"),
  q3Int = c("int", "midpoint")
)
```

## Arguments

| | |
|---|---|
| `data` | vector or dataframe with scores as numbers, or if text also provide levels |
| `levels` | optional vector with levels in order |
| `method` | optional which method to use to calculate quartiles |
| `indexMethod` | optional to indicate which type of indexing to use. Either `"sas1"` (default), `"inclusive"`, `"exclusive"`, `"sas4"`, `"excel"`, `"hl"`, `"hf8"`, or `"hf9"` |
| `q1Frac, q3Frac` | optional to indicate what type of rounding to use for each quartile. Either `"linear"` (default), `"down"`, `"up"`, `"bankers"`, `"nearest"`, `"halfdown"`, or `"midpoint"` |
| `q1Int, q3Int` | optional to indicate the use of the integer or the midpoint method for each quartile. Either `"int"` (default), or `"midpoint"`. |

## Details

To determine the quartiles a specific indexing method can be used. See `he_quartileIndexing` for details on the different methods to choose from.

Then based on the indexes either linear interpolation or different rounding methods (bankers, nearest, down, up, half-down) can be used, or the midpoint between the two values. If the index is an integer either the integer or the mid point is used. See the `he_quartilesIndex` for details on this.

Note that the rounding method can even vary per quartile, i.e. the one used for the first quartile being different than the one for the second.

I've come across the following methods:

| method | indexing | q1 integer | q1 fractional | q3 integer | q3 fractional |
|---|---|---|---|---|---|
| sas1 | sas1 | use int | linear | use int | linear |
| sas2 | sas1 | use int | bankers | use int | bankers |
| sas3 | sas1 | use int | up | use int | up |
| sas5 | sas1 | midpoint | up | midpoint | up |

| | | | | | |
|---|---|---|---|---|---|
| hf3b | sas1 | use int | nearest | use int | halfdown |
| sas4 | sas4 | use int | linear | use int | linear |
| ms | sas4 | use int | nearest | use int | halfdown |
| lohninger | sas4 | use int | nearest | use int | nearest |
| hl2 | hl | use int | linear | use int | linear |
| hl1 | hl | use int | midpoint | use int | midpoint |
| excel | excel | use int | linear | use int | linear |
| pd2 | excel | use int | down | use int | down |
| pd3 | excel | use int | up | use int | up |
| pd4 | excel | use int | halfdown | use int | nearest |
| pd5 | excel | use int | midpoint | use int | midpoint |
| hf8 | hf8 | use int | linear | use int | linear |
| hf9 | hf9 | use int | linear | use int | linear |

The following values can be used for the *method* parameter:

- inclusive = tukey =hinges = vining. (Tukey, 1977, p. 32; Siegel & Morgan, 1996, p. 77; Vining, 1998, p. 44).
- exclusive = jf. (Moore & McCabe, 1989, p. 33; Joarder & Firozzaman, 2001, p. 88).
- sas1 = parzen = hf4 = interpolated_inverted_cdf = maple3 = r4. (Parzen, 1979, p. 108; SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 363)
- sas2 = hf3 = r3. (SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 362)
- sas3 = hf1 = inverted_cdf = maple1 = r1 (SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 362)
- sas4 = hf6 = minitab = snedecor = weibull = maple5 = r6 (Hyndman & Fan, 1996, p. 363; Weibull, 1939, p. ?; Snedecor, 1940, p. 43; SAS, 1990, p. 626)
- sas5 = hf2 = CDF = averaged_inverted_cdf = r2 (SAS, 1990, p. 626; Hyndman & Fan, 1996, p. 362)
- hf3b = closest_observation
- ms (Mendenhall & Sincich, 1992, p. 35)
- lohninger (Lohninger, n.d.)
- hl1 (Hogg & Ledolter, 1992, p. 21)
- hl2 = hf5 = Hazen = maple4 = r5 (Hogg & Ledolter, 1992, p. 21; Hazen, 1914, p. ?)
- maple2
- excel = hf7 = pd1 = linear = gumbel = maple6 = r7 (Hyndman & Fan, 1996, p. 363; Freund & Perles, 1987, p. 201; Gumbel, 1939, p. ?)
- pd2 = lower
- pd3 = higher
- pd4 = nearest
- pd5 = midpoint
- hf8 = median_unbiased = maple7 = r8 (Hyndman & Fan, 1996, p. 363)
- hf9 = normal_unbiased = maple8 = r9 (Hyndman & Fan, 1996, p. 363)

*hf* is short for Hyndman and Fan who wrote an article showcasing many different methods, *hl* is short for Hog and Ledolter, *ms* is short for Mendenhall and Sincich, *jf* is short for Joarder and Firozzaman. *sas* refers to the software package SAS, *maple* to Maple, *pd* to Python's pandas library, and *r* to R.

The names *linear*, *lower*, *higher*, *nearest* and *midpoint* are all used by pandas quantile function and numpy percentile function. Numpy also uses *inverted_cdf*, *averaged_inverted_cdf*, *closest_observation*, *interpolated_inverted_cdf*, *hazen*, *weibull*, *median_unbiased*, and *normal_unbiased*.

## Value

A dataframe with:

| | |
|---|---|
| `q1` | the first (lower) quartile |
| `q3` | the third (upper/higher) quartile |
| `q1-text` | the first (lower) quartile as text (only if levels were used) |
| `q3-text` | the third (upper/higher) quartile as text (only if levels were used) |

## Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want some other descriptive measures: `me_consensus`, for the Consensus. `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_median`, for the Median. `me_quantiles`, for Quantiles. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Freund, J. E., & Perles, B. M. (1987). A new look at quartiles of ungrouped data. *The American Statistician, 41*(3), 200–203. https://doi.org/10.1080/00031305.1987.10475479

Galton, F. (1881). Report of the anthropometric committee. *Report of the British Association for the Advancement of Science, 51*, 225–272.

Gumbel, E. J. (1939). La Probabilité des Hypothèses. *Compes Rendus de l' Académie des Sciences, 209*, 645–647.

Hazen, A. (1914). Storage to be provided in impounding municipal water supply. *Transactions of the American Society of Civil Engineers, 77*(1), 1539–1640. https://doi.org/10.1061/taceat.0002563

Hogg, R. V., & Ledolter, J. (1992). *Applied statistics for engineers and physical scientists* (2nd int.). Macmillan.

Hyndman, R. J., & Fan, Y. (1996). Sample quantiles in statistical packages. *The American Statistician, 50*(4), 361–365. https://doi.org/10.2307/2684934

Joarder, A. H., & Firozzaman, M. (2001). Quartiles for discrete data. *Teaching Statistics, 23*(3), 86–89. https://doi.org/10.1111/1467-9639.00063

Langford, E. (2006). Quartiles in elementary statistics. *Journal of Statistics Education, 14*(3), 1–17. https://doi.org/10.1080/10691898.2006.11910589

Lohninger, H. (n.d.). Quartile. Fundamentals of Statistics. Retrieved April 7, 2023, from http://www.statistics4u.com/fun

McAlister, D. (1879). The law of the geometric mean. *Proceedings of the Royal Society of London, 29*(196–199), 367–376. https://doi.org/10.1098/rspl.1879.0061

Mendenhall, W., & Sincich, T. (1992). *Statistics for engineering and the sciences* (3rd ed.). Dellen Publishing Company.

Moore, D. S., & McCabe, G. P. (1989). *Introduction to the practice of statistics*. W.H. Freeman.

Parzen, E. (1979). Nonparametric statistical data modeling. *Journal of the American Statistical Association, 74*(365), 105–121. https://doi.org/10.1080/01621459.1979.10481621

SAS. (1990). SAS procedures guide: Version 6 (3rd ed.). SAS Institute.

Siegel, A. F., & Morgan, C. J. (1996). *Statistics and data analysis: An introduction* (2nd ed.). J. Wiley.

Snedecor, G. W. (1940). *Statistical methods applied to experiments in agriculture and biology* (3rd ed.). The Iowa State College Press.

Tukey, J. W. (1977). *Exploratory data analysis*. Addison-Wesley Pub. Co.

Vining, G. G. (1998). *Statistical methods for engineers*. Duxbury Press.

Weibull, W. (1939).* The phenomenon of rupture in solids*. Ingeniörs Vetenskaps Akademien, 153, 1–55.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Text dataframe
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_quartiles(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_quartiles(ex2)
```

---

## `me_quartile_range` *Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range*

### Description

There are some measures of dispersion that instead of using the full range (i.e. maximum minus minimum), make use of the quartiles. The advantage of this, is that it is less influenced by extreme values.

The Interquartile Range (Galton, 1881, p. 245) is the range how big the difference is between the third and the first quartile. If Tukey's method for the quartiles is used (*method="tukey"*), referred to as hinges, this is then also known as H-spread (Tukey, 1977, p. 44)

Yule (1911, p. 147) used half the inter-quartile range and labelled this Semi-Interquartile Range which he preferred over the term Quartile Deviation..

There is also a measure of central tendency that uses the quartiles, the Mid-Quartile (Parzen, 1980, p. 19), which is the average of the first and second quartile. It is also sometimes referred to as the Mid-Quartile Range (see for example Luo et al. (2018, p. 2), who refer to Triola, but Triola doesn't add the 'range' (Triola, 2010, p. 120))

The function uses the *me_quartiles* function and any of the methods from that function can be used.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

## Usage

```
me_quartile_range(
  data,
  levels = NULL,
  measure = c("iqr", "siqr", "qd", "mqr"),
  method = "cdf"
)
```

## Arguments

| | |
|---|---|
| `data` | vector or dataframe with scores as numbers, or if text also provide levels |
| `levels` | optional vector with levels in order |
| `measure` | the specific measure to determine. Either `"iqr"` (default), `"siqr"`, `"qd"`, or `"mqr"` |
| `method` | the method to use to determine the quartiles |

## Details

The formula used for the Interquartile Range is:

$$IQR = Q_3 - Q_1$$

This can be obtained by setting *range="iqr"*.

The IQR is mentioned in Galton (1881, p. 245) and the H-spread in Tukey (1977, p. 44).

The H-spread can be obtained by setting *range="iqr"* and *method="tukey"*.

The formula used for the Semi-Interquartile Range (Quartile Deviation) is (Yule, 1911, p. 147):

$$SIQR = \frac{Q_3 - Q_1}{2}$$

This can be obtained by setting *range="siqr"* or *range="qd"*.

The formula for the mid-quartile range used is:

$$MQR = \frac{Q_3 + Q_1}{2}$$

This can be obtained by setting *range="mqr"*.

This formula can be found in Parzen (1980, p. 19), but there are probably older references.

## Value

A dataframe with:

| | |
|---|---|
| `Q1` | the first (lower) quartile |
| `Q3` | the third (upper/higher) quartile |
| `range` | the range determined |

### Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want some other descriptive measures: `me_consensus`, for the Consensus. `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_median`, for the Median. `me_quantiles`, for Quantiles. `me_quartiles`, for Quartiles / Hinges.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Galton, F. (1881). Report of the anthropometric committee. *Report of the British Association for the Advancement of Science, 51*, 225–272.

Luo, D., Wan, X., Liu, J., & Tong, T. (2018). Optimally estimating the sample mean from the sample size, median, mid-range, and/or mid-quartile range. *Statistical Methods in Medical Research, 27*(6), 1785–1805. doi:10.1177/0962280216669183

Parzen, E. (1980). *Data modeling using quantile and density-quantile functions*. Institute of Statistics, Texas A&M University.

Triola, M. F. (2010). *Elementary statistics* (11th ed). Addison-Wesley.

Tukey, J. W. (1977). *Exploratory data analysis*. Addison-Wesley Pub. Co.

Yule, G. U. (1911). *An introduction to the theory of statistics*. Charles Griffin.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Text dataframe
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
me_quartile_range(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
me_quartile_range(ex2)
```

---

## `me_qv` *Measures of Qualitative Variation*

**Description**

The mode is the measure of central tendancy, to indicate the center for categorical data. Similar as the arithmetic mean is for numeric data. As with numeric data, the center alone is not always so informative. If your head is in a burning oven, and your feet are in a freezer, you are on average fine.

This is one of the reasons, why it is often recommended to add a measure of dispersion. It gives a clearer picture of the data, and can indicate how diverse it was (how much variation).

For categorical data there are a lot of different measures proposed, but I don't often see them being used. The most common one is probably the Variation Ratio. This is simply the percentage of cases that were not in the modal category.

The specific name of the type of measure for this qualitative variation can vary quite a lot. Some talk about dominance, differentiation, evenness, entropy, equitability, diversity, and apportionment.

I've tried to categorise the measures a bit, based on the calculations. Below is the overview of all measures available in this function.

| nr. | group | measure | source | original type |
|---|---|---|---|---|
| 1 | mode | Freeman Variation Ratio | (Freeman, 1965) | |
| 2 | mode | Berger-Parker Index | (Berger & Parker, 1970, p. 1345) | dominance |
| 3 | mode | Wilcox MODVR | (Wilcox, 1973, p. 7) | |
| 4 | mode | Wilcox RANVR | (Wilcox, 1973, p. 8) | |
| 5 | mean | Wilcox AVDEV | (Wilcox, 1973, p. 9) | |
| 6 | mean | Gibbs-Poston M4 | (Gibbs & Poston, 1975, p. 473) | differentiation |
| 7 | mean | Gibbs-Poston M5 | (Gibbs & Poston, 1975, p. 474) | differentiation |
| 8 | mean | Gibbs-Poston M6 | (Gibbs & Poston, 1975, p. 474) | differentiation |
| 9 | mean | Wilcox VARNC = | (Wilcox, 1973, p. 11) | |
| 9 | mean | Gibbs-Poston M2 = | (Gibbs & Poston, 1975, p. 472) | differentiation |
| 9 | mean | Smith-Wilson E1* | (Smith & Wilson, 1996, p. 71) | evenness |
| 10 | mean | Wilcox STDEV | (Wilcox, 1973, p. 14) | |
| 11 | entropy | Shannon-Weaver Entropy | (Shannon & Weaver, 1949, p. 20) | entropy |
| 12 | entropy | Rényi Entropy | (Rényi, 1961, p. 549) | entropy |
| 13 | entropy | Wilcox HREL = | (Wilcox, 1973, p. 16) | |
| 13 | entropy | Pielou J | (Pielou, 1966, p. 141) | diversity |
| 14 | entropy | Sheldon Index | (Sheldon, 1969, p. 467) | equitability = relative diversity |
| 15 | entropy | Heip Evenness | (Heip, 1974, p. 555) | evenness |
| 16 | evenness | Hill Diversity | (Hill, 1973, p. 428) | diversity |
| 17 | evenness | Hill Evenness | (Hill, 1973, p. 429) | evenness |
| 18 | evenness | Bulla E | (Bulla, 1994, pp. 168-169) | evenness |
| 19 | evenness | Bulla D | (Bulla, 1994, p. 169) | diversity |
| 20a | evenness | Simpson D | (Simpson, 1949, p. 688) | diversity |
| 20b | evenness | Simpson D biased | (Smith & Wilson, 1996, p. 71) | |
| 20c | evenness | Simpson D as diversity | (Wikipedia, n.d.) | |
| 20d | evenness | Simpson D as diversity biased = | (Berger & Parker, 1970, p. 1345) | |
| 20d | evenness | Gibbs-Poston M1 | (Gibbs & Poston, 1975, p. 471) | differentiation |
| 21 | evenness | Gibbs-Poston M3 | (Gibbs & Poston, 1975, p. 472) | differentiation |
| 22 | evenness | Smith-Wilson E2 | (Smith & Wilson, 1996, p. 71) | evenness |
| 23 | evenness | Smith-Wilson E3 | (Smith & Wilson, 1996, p. 71) | evenness |
| 24 | evenness | Fisher alpha | (Fisher et al., 1943, p. 55) | diversity |
| 25 | other | Wilcox MNDIF | (Wilcox, 1973, p. 9) | |
| 26 | other | Kaiser b | (Kaiser, 1968, p. 211) | apportionment |

* Smith-Wilson E1 is listed with the mean group, since it uses the average frequency. It could of course also be placed in the evenness group.

This function is shown in this YouTube video and the measures are also described at PeterStatistics.com

## Usage

```
me_qv(data, measure = "vr", var1 = 2, var2 = 1)
```

## Arguments

| | |
|---|---|
| `data` | list or dataframe |
| `measure` | optional to indicate which method to use. Either "vr" (default), "modvr", "ranvr", "avdev", "mndif", "varnc", "stdev", "hrel", "b", "m1", "m2", "m3", "m4", "m5", "m6", "d1", "d2", "d3", "d4", "bpi", "hd", "he", "swe", "re", "sw1", "sw2", "sw3", "hi", "si", "j", "b", "be", "bd", "fisher" |
| `var1` | optional additional value for some measures |
| `var2` | optional additional value for some measures |

## Details

The following measures can be determined:

- *"modvr"*, Wilcox MODVR
- *"ranvr"*, Wilcox RANVR
- *"avdev"*, Wilcox AVDEV
- *"mndif"*, Wilcox MNDIF
- *"varnc"*, Wilcox VARNC (equal to Gibbs-Poston M2 and Smith-Wilson E1)
- *"stdev"*, Wilcox STDEV
- *"hrel"*, Wilcox HREL (equal to Pielou J)
- *"m1"*, Gibbs-Poston M1
- *"m2"*, Gibbs-Poston M2 (equal to Wilcox VARNC and Smith-Wilson E1)
- *"m3"*, Gibbs-Poston M3
- *"m4"*, Gibbs-Poston M4
- *"m5"*, Gibbs-Poston M5
- *"m6"*, Gibbs-Poston M6
- *"b"*, Kaiser b
- *"bd"*, Bulla D
- *"be"*, Bulla E
- *"bpi"*, Berger-Parker index
- *"d1"*, *"d2"*, *"d3"*, *"d4"*, Simpson D and variations
- *"hd"*, Hill Diversity, requires a value for *var1*
- *"he"*, Hill Eveness, requires a value for *var1* and *var2*
- *"hi"*, Heip Index
- *"j"*, Pielou J (equal to Wilcox HREL)

- *"si"*, Sheldon Index
- *"sw1"*, Smith & Wilson E1 (equal to Wilcox VARNC and Gibbs-Poston M2)
- *"sw2"*, Smith-Wilson E2
- *"sw3"*, Smith-Wilson E3
- *"swe"*, Shannon-Weaver Entropy
- *"re"*, Renyi entropy, requires a value for *var1*
- *"vr"*, Freeman's variation ratio
- *fisher*, Fisher alpha

**MODE BASED MEASURES**

Dispersion can be seen as how much variation there is, using as a norm the center. For nominal data the measure of central tendancy is the mode, and therefor some measures of qualitative variation use the mode as the starting point.

The frequency of the modal category is then useful. This is simply the maximum of the frequencies.

**Freeman Variation Ratio** ("vr")

Perhaps one of the most popular measures of qualitative variation uses the mode. The (Freeman) Variation Ratio. It is simply the proportion of scores that do not belong to the modal category. In formula notation (Freeman, 1965, p. 41):

Formula used from Freeman (1965, p. 41):

$$v = 1 - \frac{F_{mode}}{n}$$

This variation ratio would become 0% if all cases fitted in the modal category, and all other categories don't have any cases.

A 0 (0%) would mean that all cases were in the modal category. A 1 (100%) would indicate that no cases were in the modal category. However, this seems impossible to ever occur, since the modal category is the category with the highest frequency, which is impossible to be 0, unless there are no cases at all.

**Berger–Parker index** ("bpi")

The variation ratio is the opposite of the Berger-Parker Index, which is simply the proportion of scores that did fit in the modal category. In formula notation (Berger & Parker, 1970, p. 1345):

$$BPI = \frac{F_{mode}}{n}$$

Berger and Parker refer to this as a dominance measure, to indicate how "dominant" the modal category is.

A 1 (100%) would mean that all cases were in the modal category. A 0 (0%) would indicate that no cases were in the modal category. However, this seems impossible to ever occur, since the modal category is the category with the highest frequency, which is impossible to be 0, unless there are no cases at all.

**Wilcox MODVR** ("modvr")

This looks at the difference of the frequency for each category with the modal frequency. This then gets divided by $n \times (k - 1)$ to standardize the results to 0 to 1.

It is a modification of the Freeman Variation Ratio, hence the name MODVR. Wilcox noted that the Freeman VR can never reach the maximum value of 1.

The formula used is (Wilcox, 1973, p. 7):

$$\text{MODVR} = \frac{\sum_{i=1}^{k} F_{mode} - F_i}{n \times (k-1)} = \frac{k \times F_{mode} - n}{n \times (k-1)}$$

**Wilcox RANVR** ("ranvr")

Short for 'range variation ratio' this measure is very similar to Freeman's VR. Instead of looking simply at the mode, it looks at the range.

The formula used is (Wilcox, 1973, p. 8):

$$\text{RANVR} = 1 - \frac{F_{mode} - F_{min}}{F_{mode}}$$

**MEAN BASED MEASURES**

The following measures use the average count to determine the variation. i.e.

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

**Wilcox AVDEV** ("avdev")

This simply follows the mean absolute deviation analogue but then using frequencies. Again this is then standardized.

The formula used is (Wilcox, 1973, p. 9):

$$\text{AVDEV} = 1 - \frac{\sum_{i=1}^{k} \left|F_i - \bar{F}\right|}{2 \times \frac{n}{k} \times (k-1)} = 1 - \frac{k \times \sum_{i=1}^{k} \left|F_i - \bar{F}\right|}{2 \times n \times (k-1)}$$

**Gibbs-Poston M4** ("m4")

The formula used (Gibbs & Poston, 1975, p. 473):

$$\text{M4} = 1 - \frac{\sum_{i=1}^{k} \left|F_i - \bar{F}\right|}{2 \times n}$$

**Gibbs-Poston M5** ("m5")

The problem with M4 is that it can never be 0, so to adjust for this M5 could be used but is computationally then more difficult.

The formula used (Gibbs & Poston, 1975, p. 474):

$$\text{M5} = 1 - \frac{\sum_{i=1}^{k} \left|F_i - \bar{F}\right|}{2 \times \left(n - k + 1 - \bar{F}\right)}$$

**Gibbs-Poston M6** ("m6")

The formula used (Gibbs & Poston, 1975, p. 474):

$$\text{M6} = k \times \left(1 - \frac{\sum_{i=1}^{k} \left|F_i - \bar{F}\right|}{2 \times n}\right) = k \times \text{M4}$$

**Wilcox VARNC** ("varnc"), **Gibbs-Poston M2** ("m2"), and **Smith & Wilson E1** ("sw1")

This is similar as the variance for scale variables.

The formula used is (Wilcox, 1973, p. 11):

$$\text{VARNC} = 1 - \frac{\sum_{i=1}^{k}\left(F_i - \bar{F}\right)^2}{\frac{n^2 \times (k-1)}{k}} = \frac{k \times \left(n^2 - \sum_{i=1}^{k} F_i^2\right)}{n^2 \times (k-1)}$$

This is the same as Gibbs and Poston's **M2** ("m2"). Their formula looks different but has the same result (Gibbs & Poston, 1975, p. 472)

$$\text{M2} = \frac{1 - \sum_{i=1}^{k} p_i^2}{1 - \frac{1}{k}} = \frac{\text{M1}}{1 - \frac{1}{k}} = \frac{k}{k-1} \times \text{M1}$$

It is also the same as Smith and Wilson's first evenness measure ("sw1").

The formula used (Smith & Wilson, 1996, p. 71):

$$E_1 = \frac{1 - D_s}{1 - \frac{1}{k}}$$

With $D_s$ being Simpson's D, but defined as:

$$D_s = \sum_{i=1}^{k}\left(\frac{F_i}{n}\right)^2$$

**Wilcox STDEV** ("stdev")

As with the variance for scale variables, we can take the square root to obtain the standard deviation.

The formula used can be from the VARNC or the MNDIF (Wilcox, 1973, p. 14):

$$\text{STDEV} = 1 - \sqrt{\frac{\sum_{i=1}^{k}\left(F_i - \bar{F}\right)^2}{\left(n - \bar{F}\right)^2 + (k-1)\bar{F}^2}} = 1 - \sqrt{\frac{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k}\left(F_i - F_j\right)^2}{n^2 \times (k-1)}}$$

**ENTROPY**

Entropy is sometimes referred to as the expected value of the surprise. It tells on average how surprised we might be about the outcome, and is also used as a measure with qualitative data.

I enjoyed the simple explanation on entropy from StatQuest, their video is available here.

It deals a lot with proportions rather than the counts themselves

**Shannon-Weaver Entropy** ("swe")

The formula used (Shannon & Weaver, 1949, p. 20):

$$H_{sw} = -\sum_{i=1}^{k} p_i \times \ln\left(p_i\right)$$

**Rényi entropy** ("re")

This is a generalisation for Shannon entropy.

The formula used is (Rényi, 1961, p. 549):

$$H_q = \frac{1}{1-q} \times \log_2\left(\sum_{i=1}^{k} p_i^q\right)$$

**Wilcox HREL** ("hrel") and **Pielou J** ("j")

This uses Shannon's entropy but divides it over the maximum possible uncertainty.

The formula used (Wilcox, 1973, p. 16):

$$\text{HREL} = \frac{-\sum_{i=1}^{k} p_i \times \log_2 p_i}{\log_2 k}$$

This is the same as Pielou J. ("j")

The formula used (Pielou, 1966, p. 141):

$$J = \frac{H_{sw}}{\ln\left(k\right)}$$

**Sheldon Index** ("si")

The formula used (Sheldon, 1969, p. 467):

$$E = \frac{e^{H_{sw}}}{k}$$

**Heip Index** ("hi")

The formula used is (Heip, 1974, p. 555):

$$E_h = \frac{e^{H_{sw}} - 1}{k - 1}$$

**EVENNESS and DIVERSITY**

**Hill Diversity** ("hd")

The formula used is (Hill, 1973, p. 428):

$$N_a = \begin{cases} \left(\sum_{i=1}^{k} p_i^a\right)^{\frac{1}{1-a}} & \text{if } a \neq 1 \\ e^{H_{sw}} & \text{if } = 1 \end{cases}$$

**Hill Eveness** ("he")

The formula used is (Hill, 1973, p. 429):

$$E_{a,b} = \frac{N_a}{N_b}$$

Where $N_a$ and $N_b$ are Hill's diversity values for a and b.

**Bulla E** ("be")

Bulla's evenness measure.

The formula used is (Bulla, 1994, pp. 168-169):

$$E_b = \frac{O - \frac{1}{k} - \frac{k-1}{n}}{1 - \frac{1}{k} - \frac{k-1}{n}}$$

With:

$$O = \sum_{i=1}^{k} \min\left(p_i, \frac{1}{k}\right)$$

**Bulla D** ("bd")

Bulla's Evenness measure converted to a diversity measure.

The formula used is (Bulla, 1994, p. 169):

$$D_b = E_b \times k$$

Where $E_b$ is Bulla E value.

With:

$$O = \sum_{i=1}^{k} \min\left(p_i, \frac{1}{k}\right)$$

**Simpson D** ("d1", "d2", "d3", "d4" = Gibbs-Poston M1)

The formula used is based on Simpson (1949, p. 688):

$$D_1 = \frac{\sum_{i=1}^{k} F_i \times (F_i - 1)}{n \times (n-1)}$$

Another alternative is for a population:

$$D_2 = \sum_{i=1}^{k} \left(\frac{F_i}{n}\right)^2$$

Often the result is subtracted from 1 to reverse the scale.

$$D_3 = 1 - \frac{\sum_{i=1}^{k} F_i \times (F_i - 1)}{n \times (n-1)}$$

and

$$D_4 = 1 - \sum_{i=1}^{k} \left(\frac{F_i}{n}\right)^2$$

This last one is then the same as Gibb-Poston M1 (Gibbs & Poston, 1975, p. 471):

$$\text{M1} = 1 - \sum_{i=1}^{k} p_i^2$$

**Gibbs-Poston M3** ("m3")

The formula used (Gibbs & Poston, 1975, p. 472):

$$\text{M3} = \frac{1 - \sum_{i=1}^{k} p_i^2 - p_{min}}{1 - \frac{1}{k} - p_{min}}$$

With $p_{min}$ the lowest proportion

**Smith & Wilson E2** ("sw2")

The formula used (Smith & Wilson, 1996, p. 71):

$$E_2 = \frac{\ln\left(D_s\right)}{\ln\left(k\right)}$$

With $D_s$ being Simpson's D, but defined as:

$$D_s = \sum_{i=1}^{k}\left(\frac{F_i}{n}\right)^2$$

**Smith & Wilson E3** ("sw3")

The formula used (Smith & Wilson, 1996, p. 71):

$$E_3 = \frac{1}{D_s \times k}$$

With $D_s$ being Simpson's D, but defined as:

$$D_s = \sum_{i=1}^{k}\left(\frac{F_i}{n}\right)^2$$

**Fisher alpha** ("fisher")

The formula used (Fisher et al., 1943, p. 55):

$$k = \alpha \times \ln\left(1 + \frac{n}{\alpha}\right)$$

The function uses a simple binary search to find the value for $\alpha$ such that the result of the above formula will produce the number of categories ($k$).

**OTHER***

**Wilcox MNDIF** ("mndif")

Analog of the mean difference measure for scale variables.

The formula used is (Wilcox, 1973, p. 9):

$$\text{MNDIF} = 1 - \frac{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k}\left|F_i - F_j\right|}{n \times (k-1)}$$

**Kaiser b**

The formula used (Kaiser, 1968, p. 211):

$$B = 1 - \sqrt{1 - \left(\sqrt[k]{\prod_{i=1}^{k}\frac{f_i \times k}{n}}\right)^2}$$

Kaiser also provides rules-of-thumb for interpretation (see `th_kaiser_b`, for these).

## Value

Dataframe with

| | |
|---|---|
| `value` | the value of the requested measure |
| `measure` | description of the measure calculated |
| `source` | source used for calculation |

### Before, After and Alternatives

Before this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table. `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might want to perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Berger, W. H., & Parker, F. L. (1970). Diversity of planktonic foraminifera in deep-sea sediments. *Science, 168*(3937), 1345–1347. doi:10.1126/science.168.3937.1345

Bulla, L. (1994). An index of evenness and its associated diversity measure. *Oikos, 70*(1), 167–171. doi:10.2307/3545713

Fisher, R. A., Corbet, A. S., & Williams, C. B. (1943). The relation between the number of species and the number of individuals in a random sample of an animal population. *The Journal of Animal Ecology, 12*(1), 42–58. doi:10.2307/1411

Freeman, L. C. (1965). *Elementary applied statistics: For students in behavioral science*. Wiley.

Gibbs, J. P., & Poston, D. L. (1975). The division of labor: Conceptualization and related measures. *Social Forces, 53*(3), 468. doi:10.2307/2576589

Heip, C. (1974). A new index measuring evenness. *Journal of the Marine Biological Association of the United Kingdom, 54*(3), 555–557. doi:10.1017/S0025315400022736

Hill, M. O. (1973). Diversity and evenness: A unifying notation and its consequences. *Ecology, 54*(2), 427–432. doi:10.2307/1934352

Kaiser, H. F. (1968). A measure of the population quality of legislative apportionment. *American Political Science Review, 62*(1), 208–215. doi:10.2307/1953335

Pielou, E. C. (1966). The measurement of diversity in different types of biological collections. *Journal of Theoretical Biology, 13*, 131–144. doi:10.1016/0022-5193(66)90013-0

Renyi, A. (1961). On measures of entropy and information. *Contributions to the Theory of Statistics, 1*, 547–562.

Shannon, C. E., & Weaver, W. (1949). *The mathematical theory of communication*. The university of Illinois press.

Sheldon, A. L. (1969). Equitability indices: Dependence on the species count. *Ecology, 50*(3), 466–467. doi:10.2307/1933900

Simpson, E. H. (1949). Measurement of diversity. *Nature, 163*(4148), Article 4148. doi:10.1038/163688a0

Smith, B., & Wilson, J. B. (1996). A consumer's guide to evenness indices. *Oikos, 76*(1), 70–82. doi:10.2307/3545749

Wilcox, A. R. (1973). Indices of qualitative variation and political measurement. *Political Research Quarterly, 26*(2), 325–343. doi:10.1177/106591297302600209

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
me_qv(ex1)

#Example 2: a list
ex2 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED", "DIVORCED",
"DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED", "DIVORCED",
"NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
me_qv(ex2, "swe")
```

---

## `me_variation` *Measures of Quantitative Variation*

### Description

Probably the most famous measure of dispersion is the standard deviation, but there are more. This function provides a variety of measures and allows the creation of your own version.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

### Usage

```
me_variation(
  data,
  levels = NULL,
  measure = "std",
  ddof = 1,
  center = "mean",
  azs = "square"
)
```

### Arguments

| | |
|---|---|
| `data` | : list or dataframe |
| `levels` | : dictionary, optional coding to use |
| `measure` | : "std", "var", "mad", "madmed", "medad", "stddm", "cv", "cd", "own", optional the measure to determine. Default is "std" |
| `ddof` | : float, optional option to adjust the division in standard deviation or variance with. Default is 1. |
| `center` | : "mean", "median", "mode" or float, optional if measure is "own" the value to use as center. Default is "mean" |
| `azs` | : "square", "abs", optional if measure is "own" the way to avoid a zero sum. Either by squaring or absolute value |

## Details

**Standard Deviation** (std)

The formula used is:

$$s = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n - d}}$$

Where $d$ is the offset specified at *ddof*. By default this is 1, giving the sample standard deviation.

**Variance** (var)

The formula used is:

$$s^2 = \frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n - d}$$

Where $d$ is the offset specified at *ddof*. By default this is 1, giving the sample standard deviation.

**Mean Absolute Deviation** (mad)

The formula used is:

$$MAD = \frac{\sum_{i=1}^{n}\left|x_i - \bar{x}\right|}{n}$$

**Mean Absolute Deviation from the Median** (madmed)

The formula used is:

$$MAD = \frac{\sum_{i=1}^{n}\left|x_i - \tilde{x}\right|}{n}$$

Where $\tilde{x}$ is the median

**Median Absolute Deviation** (medad)

The formula used is:

$$MAD = MED\left(\left|x_i - \tilde{x}\right|\right)$$

**Decile Standard Deviation**

The formula used is (Siraj-Ud-Doulah, 2018, p. 310):

$$s_{dm} = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - DM\right)^2}{n - d}}$$

Where DM is the decile mean.

**Coefficient of Variation** (cv)

The formula used is (Pearson, 1896, p. 277):

$$CV = \frac{s}{\bar{x}}$$

**Coefficient of Diversity** (cd)

The formula used is (Siraj-Ud-Doulah, 2018, p. 310):

$$CD = \frac{s_{dm}}{DM}$$

**Own** it's possible to create one's own method. Decide on a specific center. Default options are the mean, median and mode. Then on either to sum the squared deviations or the absolute differences.

### Value

A dataframe with:

- *value*, the value of the measure
- *measure*, description of the measure

### Before, After and Alternatives

Before this you might want to create a binned frequency table or a visualisation: `tab_frequency_bins`, to create a binned frequency table. `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

After this you might want some other descriptive measures: `me_mode_bin`, for Mode for Binned Data. `me_mean`, for different types of mean.

Or a perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Pearson, K. (1896). Contributions to the mathematical theory of evolution. III. Regression, Heredity, and Panmixia. *Philosophical Transactions of the Royal Society of London*. (A.), 1896, 253–318.

Siraj-Ud-Doulah, M. (2018). Alternative measures of standard deviation coefficient of variation and standard error. *International Journal of Statistics and Applications, 8*(6), 309–315. https://doi.org/10.5923/j.statistics.201

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
# Example 1: Numeric dataframe
ex1 = studentDf[['Gen_Age']]
me_variation(ex1)

# Example 2: Mean Absolute Deviation of a Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5)
me_variation(ex2, measure='mad')
```

---

## `ph_column_proportion` *Post-Hoc Column Proportion Test*

### Description

Post-Hoc Column Proportion Test

### Usage

```
ph_column_proportion(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  seMethod = "spss"
)
```

### Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `seMethod` | optional methdod for standard error. Either "spss" (default) or "marascuilo". |

---

## `ph_conover_iman` *Post-Hoc Conover-Iman Test*

### Description

This can be used as a post-hoc test for a Kruskal-Wallis test (see ts_kruskal_wallis()).

The test compares each possible pair of categories from the catField and their mean rank. The null hypothesis is that these are then equal. A simple Bonferroni adjustment is also made for the multiple testing.

Other post-hoc tests that could be considered are Dunn, Nemenyi, Steel-Dwass, a pairwise Mann-Whitney U, or pairwise Mood-Median.

### Usage

```
ph_conover_iman(catField, ordField, categories = NULL, levels = NULL)
```

### Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |

### Details

The formula used is (Conover & Iman, 1979, p. 11):

$$t_{1,2} = \frac{\bar{r}_1 - \bar{r}_2}{\sqrt{S^2 \times \frac{n-1-T}{n-k} \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

$$df = n - k$$

$$sig. = 1 - T\left(\left|t_{1,2}\right|, df\right)$$

With:

$$S^2 = \frac{\sum_{j=1}^{k}\sum_{i=1}^{n_j} r_{i,j}^2 - \frac{n\times(n+1)^2}{4}}{n-1}$$

$$T = \frac{\sum_{i=1}^{k}\frac{R_i^2}{n_i} - \frac{n\times(n+1)^2}{4}}{S^2}$$

$$R_i = \sum_{j=1}^{n_i} r_{i,j}$$

Note that $S^2, T, k, n$ are all based on all scores, including those not in the selected pair.

The formula can also be found in Conover (1980, pp. 230-231).

*Symbols used*

- $k$, the number of categories
- $n_i$, the number of scores in category i
- $r_{i,j}$, the rank of the j-th score in category i using all original scores (incl. those not in the comparison).
- $R_i$, the sum of the ranks in category i
- $\bar{r}_i$, the average of the ranks in category i, using all original scores (incl. those not in the comparison).
- $T\left(\dots\right)$, the cumulative distribution function of the Student t distribution.

## Value

A dataframe with:

| | |
|---|---|
| `cat. 1` | one of the two categories being compared |
| `cat. 2` | second of the two categories being compared |
| `n1` | number of cat. 1. cases in comparison |
| `n2` | number of cat. 2 cases in comparison |
| `mean rank 1` | mean rank of cases in cat. 1, based on all cases (incl. categories not in comparison) |
| `mean rank 2` | mean rank of cases in cat. 2, based on all cases (incl. categories not in comparison) |
| `statistic` | the t-value of the test |
| `df` | the degrees of freedom |
| `p-value` | the p-value (significance) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Conover, W. J. (1980). *Practical nonparametric statistics* (2nd ed.). Wiley.

Conover, W. J., & Iman, R. L. (1979). *On multiple-comparisons procedures* (LA-7677-MS; pp. 1–14). Los Alamos Scientific Laboratory.

## ph_dunn *Post-Hoc Dunn Test*

### Description

This can be used as a post-hoc test for a Kruskal-Wallis test (see ts_kruskal_wallis()).

The test compares each possible pair of categories from the catField and their mean rank. The null hypothesis is that these are then equal. A simple Bonferroni adjustment is also made for the multiple testing.

Dunn (1964) describes two procedures. The first is his own, the second is one from Steel (1960) for comparison. The difference is that in Dunn's procedure the mean rank of each category is based on the scores of all categories, including those that are not being compared, while Steel's procedure re-calculates the mean rank for each category using only the scores from the two categories being compared. This later one wouls make it very similar to a pairwise Mann-Whitney U test (see ph_mann_whitney()).

Other post-hoc tests that could be considered are Nemenyi, Steel-Dwass, Conover, a pairwise Mann-Whitney U, or pairwise Mood-Median.

### Usage

```
ph_dunn(catField, ordField, categories = NULL, levels = NULL)
```

### Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |

### Details

The formula used (Dunn, 1964, p. 249):

$$z_{1,2} = \frac{\bar{r}_1 - \bar{r}_2}{\sqrt{\sigma_m^2}}$$

$$sig. = 2 \times (1 - \Phi(z))$$

With:

$$\sigma_m^2 = \left(\frac{n \times (n+1)}{12} - \frac{T}{12 \times (n-1)}\right) \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)$$

$$T = \sum_{j=1}^{k} t_j^3 - t_j$$

$$\bar{r}_i = \frac{R_i}{n_i}$$

$$R_i = \sum_{j=1}^{n_i} r_{i,j}$$

*Symbols used*

- $k$, the number of categories
- $t_j$, the frequency of the j-th unique rank.
- $n_i$, the number of scores in category i
- $r_{i,j}$, the rank of the j-th score in category i using all original scores (incl. those not in the comparison).
- $R_i$, the sum of the ranks in category i
- $\bar{r}_i$, the average of the ranks in category i, using all original scores (incl. those not in the comparison).
- $\Phi\left(\ldots\right)$, the cumulative distribution function of the standard normal distribution.

## Value

A dataframe with:

| | |
|---|---|
| `cat. 1` | one of the two categories being compared |
| `cat. 2` | second of the two categories being compared |
| `n1` | number of cat. 1. cases in comparison |
| `n2` | number of cat. 2 cases in comparison |
| `mean rank 1` | mean rank of cases in cat. 1, based on all cases (incl. categories not in comparison) |
| `mean rank 2` | mean rank of cases in cat. 2, based on all cases (incl. categories not in comparison) |
| `statistic` | the z-value of the test |
| `p-value` | the p-value (significance) |
| `adj. p-value` | the Bonferroni adjusted p-value |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Dunn, O. J. (1964). Multiple comparisons using rank sums. *Technometrics, 6*(3), 241–252. doi:10.1080/00401706.1964.1

Steel, R. G. D. (1960). A rank sum test for comparing all pairs of treatments. *Technometrics, 2*(2), 197–207. doi:10.1080/00401706.1960.10489894

---

# `ph_dunn_q` *Post-Hoc Dunn Test (for Cochran Q test)*

## Description

An adaptation from IBM SPSS on the Dunn test, so it can be used as a post-hoc test for a Cochran Q test.

## Usage

```
ph_dunn_q(data, success = NULL)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with the binary scores |
| `success` | indicator for what is considered a success (default is first value found) |

## Details

The formula used (IBM, 2021, p. 814):

$$z_{1,2} = \frac{\bar{d}_{1,2}}{SE}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_{1,2}\right|\right)\right)$$

With:

$$\bar{d}_{1,2} = \frac{ns_1 - ns_2}{n}$$

$$SE = \sqrt{2 \times \frac{k \times \sum_{i=1}^{n} R_i - \sum_{i=1}^{n} R_i^2}{n^2 \times k \times (k-1)}}$$

$$R_i = \sum_{j=1}^{k} s_{i,j}$$

$$ns_j = \sum_{i=1}^{n} s_{i,j}$$

$$s_{i,j} = \begin{cases} 1 & \text{if } x_{i,j} = \text{success} \\ 0 & \text{if } x_{i,j} \neq \text{success} \end{cases}$$

IBM SPSS mentions this is an adaptation from Dunn (1964), originally for the Kruskal-Wallis test.

The Bonferroni adjustment is done using:

$$sig._{adj} = \min\left(sig. \times n_c, 1\right)$$

$$n_c = \frac{k \times (k-1)}{2}$$

*Symbols used*

- $x_{i,j}$, the score in row i and column j
- $k$, the number of variables
- $n$, the total number of cases used
- $ns_j$, the total number of successes in column j
- $R_i$, the total number of successes in row i
- $\Phi\left(\ldots\right)$, the standard normal cumulative distribution function.
- $n_c$, the number of comparisons (pairs)

### Value

A dataframe with:

| | |
|---|---|
| `category 1` | label of first variable in comparison |
| `category 2` | label of second variable in comparison |
| `n suc. 1` | number of successes in first variable in comparison |
| `n suc. 2*` | number of successes in second variable in comparison |
| `statistic` | test statistic |
| `z-value` | standardized test statistic (z-value) |
| `p-value` | p-value of the z-value |
| `adj. p-value` | Bonferroni corrected p-value |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Dunn, O. J. (1964). Multiple comparisons using rank sums. *Technometrics, 6*(3), 241–252. doi:10.1080/00401706.1964.1

IBM. (2021). IBM SPSS Statistics Algorithms. IBM.

---

## `ph_friedman` *Post-Hoc Tests for a Friedman Test*

### Description

A post-hoc test after a Friedman test can be used to determine which variables differ significantly.

This function provides three options: Dunn, Conover, and Nemenyi.

### Usage

```
ph_friedman(data, levels = NULL, method = "dunn", ties = TRUE)
```

### Arguments

| | |
|---|---|
| `data` | dataframe. A column for each variable |
| `levels` | vector, optional. Indication of what the levels are in order |
| `method` | string, optional. Post-Hoc method to use. Either "dunn" (default), "conover", "nemenyi" |
| `ties` | boolean, optional. Apply a ties correction. Default is True |

## Details

### Conover

Bartz-Beielstein et al. (2010, p. 319) attributes this to Conover (1999) (but also seen sites refering to Conover (1980), just different editions of the book) and uses as a formula:

$$t_{1,2} = \frac{|R_1 - R_2|}{SE}$$

$$SE = \sqrt{\frac{2 \times n \times \left(1 - \frac{\chi_F^2}{n \times (k-1)}\right) \times \left(\sum_{i=1}^{n}\sum_{j=1}^{k} r_{i,j}^2 - \frac{n \times k \times (k+1)^2}{4}\right)}{(n-1) \times (k-1)}}$$

With:

$$R_j = \sum_{i=1}^{n} r_{i,j}$$

In the original source it mentions $2 \times k$ in $SE$ instead of $2 \times n$, this was indeed an error (pers. comm. with Conover).

Gnambs (n.d.) and BrightStat (n.d.) show a different formula, that gives the same result, and is the one the function uses:

$$SE = \sqrt{\frac{2 \times \left(\sum_{i=1}^{n}\sum_{j=1}^{k} r_{i,j}^2 - \sum_{j=1}^{k} R_j^2\right)}{(n-1) \times (k-1)}}$$

The significance is then determined using:

$$sig. = 2 \times \left(1 - T\left(|t_{i,2}|, df\right)\right)$$

Note that in the calculation $SE$ is determined using all ranks, including those not in the comparison.

### Nemenyi

Pohlert (2016, p. 15) shows the formula from Nemenyi (1963) as well as in Demšar (2006, pp. 11-12):

$$q_{1,2} = \frac{|R_1 - R_2|}{\sqrt{\frac{k \times (k+1)}{6 \times n}}} \times \sqrt{2}$$

$$df = n - k$$

This follows then a studentized range distribution with:

$$sig. = 1 - Q\left(q_{1,2}, k, df\right)$$

### Dunn

Benavoli et. al (2016, pp. 2-3) and IBM SPSS (2021, p. 814):

$$z_{1,2} = \frac{|R_1 - R_2|}{SE}$$

$$SE = \sqrt{\frac{k \times (k+1)}{6 \times n}}$$

This follows a standard normal distribution:

$$sig. = 2 \times \left(1 - \Phi\left(|z_{i,2}|\right)\right)$$

### Bonferroni adjustment

The Bonferroni adjustment is done using:

$$sig._{adj} = \min\left(sig. \times n_c, 1\right)$$

$$n_c = \frac{k \times (k-1)}{2}$$

*Symbols Used*

- $n$, the number of cases
- $k$, the number of variables
- $r_{i,j}$, the rank of case i, in variable j. The ranks are determined for each case.
- $\Phi\left(\ldots\right)$, the standard normal cumulative distribution function.
- $Q\left(\ldots\right)$, the studentized range distribution cumulative distribution function.
- $T\left(\ldots\right)$, the Student t cumulative distribution function.
- $n_c$, the number of comparisons (pairs)

## Value

res : dataframe with the following columns

| | |
|---|---|
| `field 1` | label of first column in pair |
| `field 2` | label of second column in pair |
| `n` | sample size |
| `statistic` | test statistic used |
| `df` | degrees of freedom (if applicable) |
| `p-value` | the p-value (significance) |
| `adj. p-value` | Bonferroni adjusted p-value |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Benavoli, A., Corani, G., & Mangili, F. (2016). Should we really use post-hoc tests based on mean-ranks? *Journal of Machine Learning Research, 17*, 1–10. doi:10.48550/ARXIV.1505.02288

BrightStat. (n.d.). Friedman test. BrightStat. Retrieved November 5, 2023, from https://secure.brightstat.com/index.php?

Conover, W. J. (1980). *Practical nonparametric statistics* (2nd ed.). Wiley.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research, 7*, 1–30. doi:10.5555/1248547.1248548

Gnambs, T. (n.d.). SPSS Friedman. http://timo.gnambs.at/sites/default/files/spss_friedmanph.sps

IBM. (2021). IBM SPSS Statistics Algorithms. IBM.

Nemenyi, P. (1963). *Distribution-free Multiple Comparisons*. Princeton University.

Pohlert, T. (2016). The pairwise multiple comparison of mean ranks package (PMCMR). https://cran.r-hub.io/web/packages/PMCMR/vignettes/PMCMR.pdf

---

## `ph_mcnemar_co` *Post-Hoc McNemar Test - Collapsed*

---

### Description

After a (McNemar-)Bowker test a post-hoc test can potentially locate where the changes occured. This can be done use a McNemar test, which is the Bowker test but for 2x2 tables.

There are two variations, one is to simply compare each possible pair of categories (pairwise comparison), or compare each category with all other categories (collapsed comparison). This function is for the collapsed version, see **ph_mcnemar_pw()** for the pairwise version.

Instead of using the McNemar test it is also possible to use the binomial test, which will be used if exact is set to True.

### Usage

```
ph_mcnemar_co(field1, field2, categories = NULL, exact = FALSE, cc = FALSE)
```

### Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |
| `exact` | boolean, optional, use of exact binomial distribution (default is False) |
| `cc` | boolean, optional, use of a continuity correction (default is False) |

### Details

The formula used is (McNemar, 1947, p. 156):

$$\chi_M^2 = \frac{\left(F_{1,2} - F_{2,1}\right)^2}{F_{1,2} + F_{2,1}}$$

$$df = 1$$

$$sig. = 1 - \chi^2\left(\chi_M^2, df\right)$$

If a continuity correction is applied the formula changes to:

$$\chi_{M*}^2 = \frac{\left(\left|F_{1,2} - F_{2,1}\right| - 1\right)^2}{F_{1,2} + F_{2,1}}$$

The formula used for the binomial test is:

$$sig. = 2 \times \text{Bin}\left(F_{1,2} + F_{2,1}, \min\left(F_{1,2}, F_{2,1}\right), 0.5\right)$$

The formula used for the binomial test with a mid-p correction:

$$sig. = 2 \times \text{Bin}\left(F_{1,2} + F_{2,1}, \min\left(F_{1,2}, F_{2,1}\right), 0.5\right) - \text{bin}\left(F_{1,2} + F_{2,1}, \min\left(F_{1,2}, F_{2,1}\right), 0.5\right)$$

The number of pairwise tests $n_{comp}$ ) is:

$$n_{comp} = \frac{k \times (k-1)}{2}$$

The adjusted p-value is then determined using a Bonferroni correction:

$$sig._{adj} = \begin{cases} sig. \times n_{comp} & \text{if } sig. \times n_{comp} \leq= 1 \\ 1 & \text{if } sig. \times n_{comp} > 1 \end{cases}$$

*Symbols used*

- $F_{1,2}$, the observed count of cases that scored category 1 on the first variable, and another category on the second.
- $F_{2,1}$, the observed count of cases that scored another category on the first variable, and category 1 on the second.
- $\chi^2\left(\dots\right)$, the cumulative distribution function for the chi-square distribution.
- $\text{Bin}\left(\dots\right)$, the cumulative distribution function for the binomial distribution.
- $\text{bin}\left(\dots\right)$, the probability mass function for the binomial distribution.

## Value

Dataframe with:

| | |
|---|---|
| `category` | the specific category compared to all other categories |
| `n` | the sample size |
| `statistic` | the chi-squared value (if applicable) |
| `df` | the degrees of freedom used in the test (if applicable) |
| `p-value` | the significance (p-value) |
| `adj. p-value` | the Bonferroni adjusted p-value |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, 12*(2), 153–157. doi:10.1007/BF02295996

---

# `ph_mcnemar_pw` *Post-Hoc McNemar Test - Pairwise*

## Description

After a (McNemar-)Bowker test a post-hoc test can potentially locate where the changes occured. This can be done use a McNemar test, which is the Bowker test but for 2x2 tables.

There are two variations, one is to simply compare each possible pair of categories (pairwise comparison), or compare each category with all other categories (collapsed comparison). This function is for the pairwise version, see **ph_mcnemar_co()** for the collapsed version.

Instead of using the McNemar test it is also possible to use the binomial test, which will be used if exact is set to True.

## Usage

```
ph_mcnemar_pw(field1, field2, categories = NULL, exact = FALSE, cc = FALSE)
```

## Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |
| `exact` | boolean, optional, use of exact binomial distribution (default is False) |
| `cc` | boolean, optional, use of a continuity correction (default is False) |

## Details

The formula used is (McNemar, 1947, p. 156):

$$\chi_M^2 = \frac{\left(F_{1,2} - F_{2,1}\right)^2}{F_{1,2} + F_{2,1}}$$

$$df = 1$$

$$sig. = 1 - \chi^2\left(\chi_M^2, df\right)$$

If a continuity correction is applied the formula changes to:

$$\chi_{M*}^2 = \frac{\left(\left|F_{1,2} - F_{2,1}\right| - 1\right)^2}{F_{1,2} + F_{2,1}}$$

The formula used for the binomial test is:

$$sig. = 2 \times \text{Bin}\left(F_{1,2} + F_{2,1}, \min\left(F_{1,2}, F_{2,1}\right), 0.5\right)$$

The formula used for the binomial test with a mid-p correction:

$$sig. = 2 \times \text{Bin}\left(F_{1,2} + F_{2,1}, \min\left(F_{1,2}, F_{2,1}\right), 0.5\right) - \text{bin}\left(F_{1,2} + F_{2,1}, \min\left(F_{1,2}, F_{2,1}\right), 0.5\right)$$

The number of pairwise tests $n_{comp}$ ) is:

$$n_{comp} = \frac{k \times (k-1)}{2}$$

The adjusted p-value is then determined using a Bonferroni correction:

$$sig._{adj} = \begin{cases} sig. \times n_{comp} & \text{if } sig. \times n_{comp} \leq= 1 \\ 1 & \text{if } sig. \times n_{comp} > 1 \end{cases}$$

*Symbols used*

- $F_{1,2}$, the observed count of cases that scored category 1 on the first variable, and category 2 on the second.
- $F_{2,1}$, the observed count of cases that scored category 2 on the first variable, and category 1 on the second.
- $\chi^2\left(\ldots\right)$, the cumulative distribution function for the chi-square distribution.
- $\text{Bin}\left(\ldots\right)$, the cumulative distribution function for the binomial distribution.
- $\text{bin}\left(\ldots\right)$, the probability mass function for the binomial distribution.

### Value

Dataframe with:

| | |
|---|---|
| `field1` | the first category compared to the second |
| `field2` | the second category compared to the first |
| `n` | the sample size |
| `statistic` | the chi-squared value (if applicable) |
| `df` | the degrees of freedom used in the test (if applicable) |
| `p-value` | the significance (p-value) |
| `adj. p-value` | the Bonferroni adjusted p-value |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, 12*(2), 153–157. doi:10.1007/BF02295996

---

## `ph_nemenyi` *Post-Hoc Nemenyi Test*

### Description

This can be used as a post-hoc test for a Kruskal-Wallis test (see ts_kruskal_wallis()).

The test compares each possible pair of categories from the catField and their mean rank. The null hypothesis is that these are then equal.

Pohlert (2016) mentions the exact version should only be used if there are no ties, and suggest to use a chi-square alternative in case of ties. This is referred to by Zaiontz (n.d.-b) as the Schaich-Hamerle test (1984).

The ties correction is taken from Pohlert (2016).

Other post-hoc tests that could be considered are Dunn, Steel-Dwass, Conover, a pairwise Mann-Whitney U, or pairwise Mood-Median.

### Usage

```
ph_nemenyi(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  version = "auto"
)
```

### Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `version` | string, optional. version of the test to use. Either "auto" (default), "exact", "sh", "sh-ties". |

### Details

The formula used (Pohlert, 2016, p. 3):

$$q_{1,2} = \frac{\bar{r}_1 - \bar{r}_2}{\sqrt{\frac{n\times(n+1)}{24} \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

$$sig. = 1 - Q\left(q_{1,2}, k, df = \infty\right)$$

A chi-square distribution can also be used. Zaiontz (n.d.) and BrightStat (n.d.) refer to this as the Schaich-Hamerle test.

The formula used then changes to (Sachs, 1982, p. 549):

$$\chi^2_{1,2} = \frac{\left(\bar{r}_1 - \bar{r}_2\right)^2}{\frac{n\times(n+1)}{12} \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}$$

$$df = k - 1$$

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

A ties correction found in Pohlert (2016, p. 3) adjusts this to:

$$\chi^2_{1,2} = \frac{\left(\bar{r}_1 - \bar{r}_2\right)^2}{(1 - T) \times \frac{n\times(n+1)}{12} \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}$$

$$T = \frac{\sum t_i^3 - t_i}{n^3 - n}$$

The original formula is most likely from Nemenyi (1963) and the Schaich and Hamerle (1984).

*Symbols used*

- $k$, the number of categories
- $t_j$, the frequency of the j-th unique rank.
- $n$, the total sample size, of all scores, incl. those not in the comparison
- $n_i$, the number of scores in category i
- $r_{i,j}$, the rank of the j-th score in category i, using all original scores (incl. those not in the comparison).
- $\bar{r}_i$, the average of the ranks in category i, using all original scores (incl. those not in the comparison).
- $Q\left(\dots\right)$, the cumulative distribution function of the standardized range distribution.
- $\chi^2\left(\dots\right)$, the cumulative distribution function of the chi-square distribution.

### Value

A dataframe with:

| | |
|---|---|
| `cat. 1` | one of the two categories being compared |
| `cat. 2` | second of the two categories being compared |
| `n1` | number of cat. 1. cases in comparison |
| `n2` | number of cat. 2 cases in comparison |
| `mean rank 1` | mean rank of cases in cat. 1, based on all cases (incl. categories not in comparison) |
| `mean rank 2` | mean rank of cases in cat. 2, based on all cases (incl. categories not in comparison) |
| `se` | the standard error used |
| `statistic` | the z-value of the test |
| `p-value` | the p-value (significance) |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

BrightStat. (n.d.). Kruskal-Wallis test. BrightStat. Retrieved October 25, 2023, from https://secure.brightstat.com/index.p

Nemenyi, P. (1963). *Distribution-free Multiple Comparisons*. Princeton University.

Pohlert, T. (2016). The pairwise multiple comparison of mean ranks package (PMCMR). https://cran.r-hub.io/web/packages/PMCMR/vignettes/PMCMR.pdf

Sachs, L. (1982). *Applied statistics: A handbook of techniques*. Springer-Verlag.

Schaich, E., & Hamerle, A. (1984). *Verteilungsfreie statistische Prüfverfahren*. Springer. doi:10.1007/978-3-642-70032-3

Zaiontz, C. (n.d.). Schaich-Hamerle Test after KW. Real Statistics Using Excel. Retrieved October 25, 2023, from https://real-statistics.com/one-way-analysis-of-variance-anova/kruskal-wallis-test/schaich-hamerle-test/

---

## `ph_pairwise_bin` *Pairwise Binary Test for Post-Hoc Analysis*

### Description

This function will perform a one-sample binary test for each possible pair in the data. This could either be a binomial, Wald or score test.

The unadjusted p-values and Bonferroni adjusted p-values are both determined.

### Usage

```
ph_pairwise_bin(
  data,
  test = "binomial",
  expCount = NULL,
  mtc = "bonferroni",
  ...
)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores |
| `test` | "binomial", "score", "wald", optional test to use for each pair |
| `expCount` | optional dataframe with categories and expected counts |
| `mtc` | optional string. Any of the methods available in p_adjust() to correct for multiple tests |
| `...` | optional additional arguments for the specific test that are passed along. |

## Value

a dataframe with:

| | |
|---|---|
| `category 1` | the label of the first category |
| `category 2` | the label of the second category |
| `n1` | the sample size of the first category |
| `n2` | the sample size of the second category |
| `n pair` | the sample size of of the pair |
| `obs. prop. 1` | the proportion in the sample of the first category |
| `exp. prop. 1` | the expected proportion for the first category |
| `statistic` | the test statistic |
| `p-value` | the unadjusted significance |
| `adj. p-value` | the adjusted significance |
| `test` | description of the test used |

## Before, After and Alternatives

Before this an omnibus test might be helpful: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

After this you might want to add an effect size measure: `es_post_hoc_gof` for various effect sizes.

Alternative post-hoc tests: `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin` for Residuals Tests using Binary Tests `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

The binary test that is performed on each pair: `ts_binomial_os` for One-Sample Binomial Test. `ts_score_os` for One-Sample Score Test. `ts_wald_os` for One-Sample Wald Test.

More info on the adjustment for multiple testing: `p_adjust`, various adjustment methods.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
# Examples: get data
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
gssDf <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = gssDf['mar1']

#Example 1 using default settings (one-sample binomial tests with equal-distance method)
ph_pairwise_bin(ex1)

#Example 2 using a score test with Yates correction:
ph_pairwise_bin(ex1, test="score", mtc='holm', cc='yates')
```

---

# `ph_pairwise_gof` *Pairwise Goodness-of-Fit Tests for Post-Hoc Analysis*

## Description

This function will perform a goodness-of-fit test for each possible pair in the data. This could be any of the goodness-of-fit tests, e.g. a Pearson chi-square.

The unadjusted p-values and Bonferroni adjusted p-values are both determined.

## Usage

```
ph_pairwise_gof(
  data,
  test = "pearson",
  expCount = NULL,
  mtc = "bonferroni",
  ...
)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores |
| `test` | "pearson", "freeman-tukey", "freeman-tukey-read", "g", "mod-log-g", "neyman", "powerdivergence", "multinomial", optional test to use for each pair |
| `expCount` | optional dataframe with categories and expected counts |
| `mtc` | optional string. Any of the methods available in p_adjust() to correct for multiple tests |
| `...` | optional additional arguments for the specific test that are passed along. |

## Value

a dataframe with:

| | |
|---|---|
| `category 1` | the label of the first category |
| `category 2` | the label of the second category |
| `n1` | the sample size of the first category |
| `n2` | the sample size of the second category |

| | |
|---|---|
| `obs. prop. 1` | the observed proportion of category 1 |
| `exp. prop. 1` | the expected proportion of category 1 |
| `statistic` | the chi-square test statistic |
| `df` | the degrees of freedom |
| `p-value` | the unadjusted significance |
| `adj. p-value` | the adjusted significance |
| `minExp` | the minimum expected count |
| `propBelow5` | the proportion of cells with an expected count below 5 |
| `test` | description of the test used |

In case of a multinomial test, the same columns except there are no *minExp* and *propBelow5* columns and:

| | |
|---|---|
| `p obs` | instead of *statistic*, showing the probability of the observed sample table |
| `n combs.` | instead of *df*, showing the number of possible tables |

## Before, After and Alternatives

Before this an omnibus test might be helpful, these are also the tests used on each pair: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

After this you might want to add an effect size measure: `es_post_hoc_gof` for various effect sizes.

Alternative post-hoc tests: `ph_pairwise_bin` for Pairwise Binary Tests. `ph_residual_gof_bin` for Residuals Tests using Binary Tests `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

More info on the adjustment for multiple testing: `p_adjust`, various adjustment methods.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
gssDf <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = gssDf['mar1']

# Example 1 using default settings
ph_pairwise_gof(ex1)

#Example 2 using a G test with Pearson correction:
ph_pairwise_gof(ex1, test="g", mtc='holm', cc='pearson')
```

## `ph_pairwise_is` *Post-Hoc Pairwise Independent Samples Test*

### Description

This function can perform various pairwise independent samples tests, for use after a one-way ANOVA, to determine which categories significantly differ from each other.

A simple Bonferroni correction is also applied.

The independent samples tests that can be used are:

- Student t, see ts_student_t_is() for details. An alternative version for this is available by using the ph_pairwise_t() function.
- Welch t, see ts_welch_t_is() for details
- Trimmed Mean / Yuen, see ts_trimmed_mean_is() for details
- Z, see ts_z_is() for details

### Usage

```
ph_pairwise_is(
  nomField,
  scaleField,
  categories = NULL,
  isTest = "student",
  trimProp = 0.1
)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |
| `isTest` | string, optional. The independent samples test to use. Either "student" (default), "welch", "trimmed", "yuen", "z" |
| `trimProp` | float, optional. The trim proportion to use, if applicable. Default is 0.1. |

### Details

The Bonferroni adjustment is simply:

$$p_{adj} = \min\left(p \times n_{comp}, 1\right)$$

$$n_{comp} = \frac{k \times (k-1)}{2}$$

*Symbols used:*

- $n_{comp}$, number of comparisons (pairs)
- $k$, number of categories

## Value

A dataframe with:

| | |
|---|---|
| `category 1` | the first category in the pair |
| `category 2` | the second category in the pair |
| `n1` | sample size of first category |
| `n2` | sample size of second category |
| `mean 1` | arithmetic mean of scores in first category |
| `mean 2` | arithmetic mean of scores in second category |
| `sample diff.` | difference between the two arithmetic means |
| `hyp diff.` | the hypothesized difference |
| `statistic` | the test-statistic |
| `df` | the degrees of freedom |
| `p-value` | the unadjusted p-value (significance) |
| `adj. p-value` | the Bonferroni adjusted p-values |
| `test` | description of test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

# `ph_pairwise_iso` *Post-Hoc Pairwise Independent Samples Test for Ordinal*

## Description

The Mann-Whitney U, Mood Median and Fligner-Policello test are designed for two categories and an ordinal variable. These can therefor be used as a post-hoc test for a Kruskal-Wallis test (see ts_kruskal_wallis()). The test compares each possible pair of categories from the catField and their mean rank. The null hypothesis is that these are then equal. A simple Bonferroni adjustment is also made for the multiple testing. Other post-hoc tests that could be considered are Dunn, Nemenyi, Steel-Dwass, Conover-Iman.

## Usage

```
ph_pairwise_iso(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  iso_test = "mann-whitney",
  ...
)
```

## Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `iso_test` | string, optional. The test of independence to use. Options are "mann-whitney" (default), "mood", "fligner-policello |
| `...` | other, optional. other arguments to pass on for the specific test used. |

## Details

This function selects each possible pair of categories and then simply runs the requested test, using only those two categories.

See ts_mann_whitney(), ts_mood_median() and/or ts_fligner_policello() for details of the calculations.

The Bonferroni adjustment is simply:

$$p_{adj} = \min\left(p \times n_{comp}, 1\right)$$

$$n_{comp} = \frac{k \times (k-1)}{2}$$

*Symbols used:*

- $n_{comp}$, number of comparisons (pairs)
- $k$, number of categories

## Value

A dataframe with:

| | |
|---|---|
| `category 1` | one of the two categories being compared |
| `category 2` | second of the two categories being compared |
| `statistic` | the test statistic |
| `df` | he degrees of freedom, if applicable |
| `p-value` | the p-value (significance) |
| `adj. p-value` | the Bonferroni adjusted p-value |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

## `ph_pairwise_ps` *Post-Hoc Pairwise Paired Samples Tests*

---

### Description

This function simply performs pairwise paired tests: Sign, Wilcoxon and Trinomial. It then adds a Bonferroni correction.

These could be used with a Friedman test, but other post-hoc tests are also available in the ph_friedman() function (a Dunn, Nemenyi and Conover test).

### Usage

```
ph_pairwise_ps(
  data,
  levels = NULL,
  test = "sign",
  appr = "wilcoxon",
  noDiff = "wilcoxon",
  ties = TRUE,
  cc = FALSE
)
```

### Arguments

| | |
|---|---|
| `data` | dataframe. A column for each variable |
| `levels` | vector, optional. Indication of what the levels are in order |
| `test` | string, optional. Test to use in pairwise comparisons. Either "sign" (details), "wilcoxon", "trinomial". |
| `appr` | string, optional. Option for sign and wilcoxon test. Default for wilcoxon is wilcoxon, for sign is appr. Either "exact", "appr", "wilcoxon", "imanz", "imant" |
| `noDiff` | string, optional. Method to deal with scores equal to mu. Either "wilcoxon" (default), "pratt", "zsplit". Only applies if test="wilcoxon" |
| `ties` | boolean, optional. Apply a ties correction. Default is True |
| `cc` | boolean, optional. use a continuity correction. Default is False. Only applies if test="wilcoxon" |

### Details

This function creates each possible pair of the variables (columns) and then uses the requested paired samples test.

See for the calculations:

- Sign test -> ts_sign_ps()
- Wilcoxon signed rank test -> ts_wilcoxon_ps()
- Trinomial test -> ts_trinomial_ps()

The Bonferroni adjustment is done using:

$$sig._{adj} = \min\left(sig. \times n_c, 1\right)$$

$$n_c = \frac{k \times (k-1)}{2}$$

Where $n_c$ is the number of comparisons (pairs)

## Value

res, a dataframe with the test results and:

| | |
|---|---|
| `var 1` | the name of the first variable in the pair |
| `var 2` | the name of the second variable in the pair |
| `adj. p-value` | the Bonferroni adjusted p-value |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

# `ph_pairwise_t` *Post-Hoc Pairwise Student T*

## Description

This function performs pairwise independent samples Student t tests, for use after a one-way ANOVA, to determine which categories significantly differ from each other.

It differs slightly in the calculation of the standard error, than the version used by using ph_pairwise_is(nomField, scaleField, isTest = "student"). This version appears to be producing the same results as SPSS shows, when using a Bonferroni correction. SPSS refers to Winer (1962) for their procedures.

A simple Bonferroni correction is also applied.

## Usage

```
ph_pairwise_t(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used:

$$t_{1,2} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{MS_w \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

$$df_w = n - k$$

$$sig. = 2 \times \left(1 - T\left(\left|t_{1,2}\right|, df_w\right)\right)$$

With:

$$MS_w = \frac{SS_w}{df_w}$$

$$SS_w = \sum_{j=1}^{k}\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols used*

- $x_{i,j}$, the i-th score in category j
- $n$, the total sample size
- $n_j$, the number of scores in category j
- $k$, the number of categories
- $\bar{x}_j$, the mean of the scores in category j
- $MS_w$, the mean square within
- $SS_w$, the sum of squares of within (sum of squared deviation of the mean)
- $df_w$, the degrees of freedom of within

The Bonferroni adjustment is simply:

$$p_{adj} = \min\left(p \times n_{comp}, 1\right)$$

$$n_{comp} = \frac{k \times (k-1)}{2}$$

*Symbols used:*

- $n_{comp}$, number of comparisons (pairs)
- $k$, number of categories

## Value

A dataframe with:

| | |
|---|---|
| `category 1` | the first category in the pair |
| `category 2` | the second category in the pair |
| `n1` | sample size of first category |
| `n2` | sample size of second category |
| `mean 1` | arithmetic mean of scores in first category |
| `mean 2` | arithmetic mean of scores in second category |

| | |
|---|---|
| `sample diff.` | difference between the two arithmetic means |
| `hyp diff.` | the hypothesized difference |
| `statistic` | the test-statistic |
| `df` | the degrees of freedom |
| `p-value` | the unadjusted p-value (significance) |
| `adj. p-value` | the Bonferroni adjusted p-values |
| `test` | description of test used |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Winer, B. J. (1962). *Statistical principles in experimental design*. McGraw Hill.

---

## `ph_residual` *Post-Hoc Residual Test*

### Description

Post-Hoc Residual Test

### Usage

```
ph_residual(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  residual = "adjusted"
)
```

### Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `residual` | optional methdod for residual to test. Either "adjusted" (default) or "standardized". |

---

## `ph_residual_gof_bin` *Post-Hoc Residuals Using Binary Tests for GoF*

---

### Description

This function will perform a residuals post-hoc test for each of the categories in a nominal field. This could either be a z-test using the standardized residuals, the adjusted residuals, or any of the one-sample binary tests.

The unadjusted p-values and Bonferroni adjusted p-values are both determined.

### Usage

```
ph_residual_gof_bin(
  data,
  test = "std-residual",
  expCount = NULL,
  mtc = "bonferroni",
  ...
)
```

### Arguments

| | |
|---|---|
| `data` | dataframe with scores |
| `test` | "adj-residual", "std-residual", "binomial", "wald", "score" optional test to use |
| `expCount` | optional dataframe with categories and expected counts |
| `mtc` | optional string. Any of the methods available in p_adjust() to correct for multiple tests |
| `...` | optional additional parameters to be passed to the test |

### Details

The formula used is for the adjusted residual test:

$$z = \frac{F_i - E_i}{\sqrt{E_i \times \left(1 - \frac{E_i}{n}\right)}}$$

$$sig = 2 \times \left(1 - \Phi\left(\left|z\right|\right)\right)$$

The formula used for the standardized residual test:

$$z = \frac{F_i - E_i}{\sqrt{E_i}}$$

$$sig = 2 \times \left(1 - \Phi\left(\left|z\right|\right)\right)$$

With:

- $F_i$, the observed count for category \$i\$
- $E_i$, the expected count for category \$i\$
- $\Phi\left(\dots\right)$, the cumulative distribution function of the standard normal distribution

If no expected counts are provide it is assumed they are all equal for each category, i.e. $E_i = \frac{n}{k}$

The Bonferroni adjustment is calculated using:

$$p_{adj} = \min\left(p \times k, 1\right)$$

The other tests use the formula from the one-sample test variant, using the expected count/n as the expected proportion.

The adjusted residuals will gave the same result as using a one-sample score test. Some sources will also call these adjusted residuals as standardized residuals (Agresti, 2007, p. 38), and the standardized residuals used in this function as Pearson residuals (R, n.d.). Haberman (1973, p. 205) and Sharpe (2015, p. 3) are sources for the terminology used in this function.

## Value

a dataframe with:

| | |
|---|---|
| `category` | the label of the first category |
| `obs. count` | the observed count |
| `exp. count` | the expected count |
| `statistic` | the test statistic |
| `p-value` | the unadjusted significance |
| `adj. p-value` | the adjusted significance |
| `test` | description of the test used |

## Before, After and Alternatives

Before this an omnibus test might be helpful: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

After this you might want to add an effect size measure: `es_post_hoc_gof` for various effect sizes.

Alternative post-hoc tests: `ph_pairwise_bin` for Pairwise Binary Tests. `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_gof` for Residuals Using Goodness-of-Fit Tests

The binary test that is performed on each category: `ts_binomial_os` for One-Sample Binomial Test. `ts_score_os` for One-Sample Score Test. `ts_wald_os` for One-Sample Wald Test.

More info on the adjustment for multiple testing: `p_adjust`, various adjustment methods.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Agresti, A. (2007). *An introduction to categorical data analysis* (2nd ed.). Wiley-Interscience. Haberman, S. J. (1973). The analysis of residuals in cross-classified tables. *Biometrics, 29*(1), 205–220. doi:10.2307/2529686 R. (n.d.). Chisq.test [Computer software]. https://stat.ethz.ch/R-manual/R-devel/library/stats/html/chisq.test.html Sharpe, D. (2015). Your chi-square test is statistically significant: Now what? Practical Assessment, *Research & Evaluation, 20*(8), 1–10.

## Examples

```
# Examples: get data
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
gssDf <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = gssDf['mar1']

#Example 1 using default settings
ph_residual_gof_bin(ex1)

#Example 2 using a binomial test and Holm correction:
ph_residual_gof_bin(ex1, test="binomial", mtc='holm')
```

---

# `ph_residual_gof_gof` *Post-Hoc Residuals Using GoF for GoF*

## Description

This function will perform a goodness-of-fit test using each category and collapsing the other categories.

The unadjusted p-values and Bonferroni adjusted p-values are both determined.

## Usage

```
ph_residual_gof_gof(
  data,
  test = "pearson",
  expCount = NULL,
  mtc = "bonferroni",
  ...
)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores |
| `test` | "pearson", "freeman-tukey", "freeman-tukey-read", "g", "mod-log-g", "neyman", "powerdivergence", "multinomial" optional test to use |
| `expCount` | optional dataframe with categories and expected counts |
| `mtc` | optional string. Any of the methods available in p_adjust() to correct for multiple tests |
| `...` | optional additional parameters to be passed to the test |

## Value

a dataframe with:

| | |
|---|---|
| `category` | the label of the first category |
| `obs. count` | the observed count of the category |
| `exp. count` | the expected count of the category |
| `statistic` | the chi-square test statistic |

| | |
|---|---|
| `df` | the degrees of freedom |
| `p-value` | the unadjusted significance |
| `adj. p-value` | the adjusted significance |
| `minExp` | the minimum expected count |
| `propBelow5` | the proportion of cells with an expected count below 5 |
| `test` | description of the test used |

In case of a multinomial test, the same columns except there are no *minExp* and *propBelow5* columns and:

| | |
|---|---|
| `p obs` | instead of *statistic*, showing the probability of the observed sample table |
| `n combs.` | instead of *df*, showing the number of possible tables |

## Before, After and Alternatives

Before this an omnibus test might be helpful, these are also the tests used on each category: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_pearson_gof`, for Pearson Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

After this you might want to add an effect size measure: `es_post_hoc_gof` for various effect sizes.

Alternative post-hoc tests: `ph_pairwise_bin` for Pairwise Binary Tests. `ph_pairwise_gof` for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin` for Residuals Tests using Binary Tests

More info on the adjustment for multiple testing: `p_adjust`, various adjustment methods.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
# Examples: get data
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
gssDf <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = gssDf['mar1']

#Example 1 using default settings
ph_residual_gof_gof(ex1)

#Example 2 using a G test and Holm correction:
ph_residual_gof_gof(ex1, test="g", mtc='holm')
```

---

`ph_sdcf` *Post-Hoc Steel-Dwass-Critchlow-Fligner Test*

---

## Description

This can be used as a post-hoc test for a Kruskal-Wallis test (see ts_kruskal_wallis()).

The test compares each possible pair of categories from the catField and their mean rank. The null hypothesis is that these are then equal.

Other post-hoc tests that could be considered are Dunn, Nemenyi, Conover, a pairwise Mann-Whitney U, or pairwise Mood-Median.

Unlike the Dunn, Nemenyi and Conover-Iman test, this test re-calculates the mean ranks for each pair, using only the scores from the two categories.

## Usage

```
ph_sdcf(catField, ordField, categories = NULL, levels = NULL)
```

## Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |

## Details

The formula used (Hollander & Wolfe, 1999, p. 241):

$$q_{1,2} = \frac{|R_1 - E_1|}{\sqrt{\sigma^2}}$$

With:

$$R_1 = \sum_{i=1}^{n_1} r_{i,1}$$

$$n_{1,2} = n_1 + n_2$$

$$E_1 = \frac{n_1 \times (n_{1,2} + 1)}{2}$$

$$\sigma^2 = \frac{n_1 \times n_2}{12} \times \left(n_{1,2} + 1 - \frac{T}{n_{1,2} - 1}\right)$$

$$T = \sum t_j^3 - t_j$$

The p-value is then determined using (Critchlow & Fligner, 1991, p. 131):

$$sig. = 1 - Q\left(q_{1,2}, k, df = \infty\right)$$

Note that while looking at the R-code for this, posted by Shigenobu (n.d.), who references Nagata and Yoshida (1997), an alternative but same result equation for the variance can be used:

$$\sigma^2 = \frac{n_1 \times n_2}{n_{1,2} \times \left(n_{1,2} - 1\right)} \times \left(\sum_{i=1}^{n_1} r_{i,1}^2 + \sum_{i=1}^{n_2} r_{i,2}^2 - \frac{n_{1,2} \times \left(n_{1,2} + 1\right)^2}{4}\right)$$

.

Steel (1960) and Dwass (1960) independently derived the basics for this test. Critchlow and Fligner (1991) added the case for larger samples using the Tukey Range Distribution, and in Hollander and Wolfe (1999) the version used here can be found, which includes a ties correction.

*Symbols used*

- $k$, the number of categories
- $t_j$, the frequency of the j-th unique rank.
- $n_i$, the number of scores in category i
- $r_{i,j}$, the rank of the j-th score in category i using only the scores from the two categories in the comparison.
- $Q\left(\dots\right)$, the cumulative distribution function of the standardized range distribution.

## Value

A dataframe with:

| | |
|---|---|
| `cat. 1` | one of the two categories being compared |
| `cat. 2` | second of the two categories being compared |
| `n1` | number of cat. 1. cases in comparison |
| `n2` | number of cat. 2 cases in comparison |
| `mean rank 1` | mean rank of cases in cat. 1, based on all cases (incl. categories not in comparison) |
| `mean rank 2` | mean rank of cases in cat. 2, based on all cases (incl. categories not in comparison) |
| `statistic` | the q-value of the test |
| `std. statistic` | the standardized q value |
| `p-value` | the p-value (significance) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Critchlow, D. E., & Fligner, M. A. (1991). On distribution-free multiple comparisons in the one-way analysis of variance. *Communications in Statistics - Theory and Methods, 20*(1), 127–139. doi:10.1080/03610929108830487

Dwass, M. (1960). *Some k-sample rank-order tests*. In I. Olkin, S. G. Ghurye, W. Hoeffding, W. G. Madow, & H. B. Mann (Eds.), Contributions to probability and statistics; Essays in honor of Harold Hotelling. Stanford University Press.

Hollander, M., & Wolfe, D. A. (1999). *Nonparametric statistical methods* (2nd ed.). Wiley.

Nagata, Y., & Yoshida, M. (1997). *The Basics of Multiple Comparisons in Statistics*. Scientist Co.

Shigenobu. (2004, July 28). Multiple comparisons using the Steel-Dwass method. http://aoki2.si.gunma-u.ac.jp/R/Steel-Dwass.html

Steel, R. G. D. (1960). A rank sum test for comparing all pairs of treatments. Technometrics, 2(2), 197–207. doi:10.1080/00401706.1960.10489894

---

## `p_adjust` *P-Value Adjustments for Multiple Testing*

### Description

Various methods exist to counter a problem with multiple testing. Bonferroni, Šidák, Hommel, Holm, Holm-Šidák, and Hochberg all attempt to control the family wise error rate (FWER), while Benjamini-Hochberg and Benjamini-Yekutieli attempt to control the false discovery rate (FDR).

FWER methods want to minimize the chance of making at least one Type I error (incorrectly rejecting the null hypothesis), while FDR methods attempt to balance the false positive and false negatives.

### Usage

```
p_adjust(p_values, method = "bonferroni", alpha = 0.05)
```

### Arguments

| | |
|---|---|
| `p_values` | list with the various p-values |
| `method` | 'bonferroni', 'sidak', 'hommel', 'holm', 'holm-sidak', 'hochberg', 'bh', 'by', 'hommel-original', 'none' optional method to use for adjustment, Default is 'bonferroni' |
| `alpha` | : float, optional alpha level to use, only applies to 'hommel-original'. Default is 0.05. |

### Details

**none** simply returns the provided p-values

**Bonferroni**

The formula used for the Bonferroni adjustment:

$$\tilde{p}_i = \min\left(1, p_i \times k\right)$$

Where $p_i$ is the p-value of test $i$, and $k$ the number of tests.

Dunn (1961, p. 53) uses the Bonferroni inequality to adjust confidence intervals, which is why this is also called the Dunn-Bonferroni adjustments.

Bonferroni describes these inequalities in two papers (1935, 1936), but unfortunately I do not read Italian. The term 'Bonferroni inequalities' can already be found in Feller (1950, p. 75)

**Šidák**

The formula used (Šidák, 1967, p. 629):

$$\tilde{p}_i = \min\left(1, 1 - \left(1 - p_i\right)^k\right)$$

Where $p_i$ is the p-value of test $i$, and $k$ the number of tests.

**Hommel**

The algorithm used (Wright, 1992, p. 1013):

1. Set $a_i = p_i$ for all $i$
2. For each $m = k, (k-1), ..., 2$ (i.e. in descending order) do the following: 2.1. For $i > (k-m)$ * 2.1.1. calculate $c_i = \frac{m \times p_i}{m+i-k}$ * 2.1.2. set $c_{min} = \min\left(c_i, for all i\right)$ * 2.1.3. if $a_i < c_{min}$ then $a_i = c_m in$ 2.2. For $i \leq (k-m)$ * 2.2.1. let $c_i = \min\left(c_{min}, m \times p_i\right)$ * 2.2.2. if $a_i < c_i$ then $a_i = ci$
3. $p_i^{adj} = a_i$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in ascending order, and $k$ the number of tests.

Hommel (1988) original procedure is I think slightly different and implemented in 'hommel-original'. The method from Wright seems to be the method used in the multipletests() function from the Python library statsmodels.stats.multitest, and the p.adjust() function from R's stats library. The advantage of Wright's algorithm, is that it doesn't require the alpha level to be known to adjust the p-values.

**Holm**

The formula used (SAS, n.d.):

$$\tilde{p}_i = \begin{cases} k \times p_1 & i = 1 \\ \max\left(\tilde{p}_{i-1}, p_i \times (k+1-i)\right) & i = 2, \dots, k \end{cases}$$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in ascending order, and $k$ the number of tests.

Holm (1979, p. 67) describes this procedure, but uses alpha level.

**Holm-Šidák**

The formula used (SAS, n.d.):

$$\tilde{p}_i = \begin{cases} 1 - \left(1 - p_1\right)^k & i = 1 \\ \max\left(\tilde{p}_{i-1}, 1 - \left(1 - p_i\right)^{k-i+1}\right) & i = 2, \dots, k \end{cases}$$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in ascending order, and $k$ the number of tests.

This uses Holm (1979, p. 67) step-down approach, but instead of using the Bonferroni adjustment, it uses Šidák.

**Hochberg**

The formula used (SAS, n.d.):

$$\tilde{p}_i = \begin{cases} p_i & i = 1 \\ \min\left(\tilde{p}_{i-1}, i \times p_i\right) & i = 2, \dots, k \end{cases}$$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in DESCENDING order, and $k$ the number of tests.

The procedure is described by Hochberg (1988, p. 801) using alpha levels for the criteria.

**Benjamini-Hochberg**

The algorithm used (Benjamini & Hochberg, 1995, p. 293):

1. Sort the p-values in ascending order
2. find $j = \max\left(i|p_i \times \frac{k}{i} \leq \alpha\right)$
3. All tests $i \leq j$ are considered significant.

To find the adjusted p-values (in reverse order):

$$\tilde{p}_i = \begin{cases} p_k & i = k \\ \min\left(\tilde{p}_{i+1}, p_i \times \frac{k}{i}\right) & i = (k-1), \dots, 1 \end{cases}$$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in ascending order, and $k$ the number of tests.

**Benjamini-Yekutieli**

The algorithm used (Benjamini & Yekutieli, 2001, p. 1169):

1. Sort the p-values in ascending order
2. Determine $C(k) = \sum_{i=1}^{k} \frac{1}{i}$
3. find $j = \max\left(i|p_i \times \frac{k \times C(k)}{i} \leq \alpha\right)$
4. All tests $i \leq j$ are considered significant.

To find the adjusted p-values (in reverse order):

$$\tilde{p}_i = \begin{cases} p_k \times C(k) & i = k \\ \min\left(\tilde{p}_{i+1}, p_i \times \frac{k \times C(k)}{i}\right) & i = (k-1), \dots, 1 \end{cases}$$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in ascending order, and $k$ the number of tests.

**Hommel Original**

Hommel (1988, p. 384) describes the following algorithm:

1. Compute $j = \max\left(i \in 1, \dots, k|p_{k-i+j} > j \times \frac{\alpha}{i} for j = 1, \dots, i\right)$
2. If the maximum does not exist, reject all, otherwise reject all with $p_i \leq \frac{\alpha}{j}$

Where $p_i$ is the p-value of test $i$ after sorting all p-values in ascending order, and $k$ the number of tests.

The function will adjust the p-values using:

$$\tilde{p}_i = \begin{cases} \min\left(1, j \times p_i\right) & j exists \\ 1 & j does not exist \end{cases}$$

## Value

p_adj_val : list with the adjusted p-values

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B: Statistical Methodology, 57*(1), 289–300. doi:10.1111/j.2517-6161.1995.tb02031.x

Benjamini, Y., & Yekutieli, D. (2001). The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics, 29*(4), 1165–1188. doi:10.1214/aos/1013699998

Bonferroni, C. E. (1935). Il calcolo delle assicurazioni su gruppi di teste. In *Studi in Onore del Professore Salvatore Ortu Carboni* (pp. 13–60).

Bonferroni, C. E. (1936). *Teoria statistica delle classi e calcolo delle probabilità*. Pubblicazioni Del R Istituto Superiore Di Scienze Economiche e Commerciali Di Firenze, 8, 3–62.

Dunn, O. J. (1961). Multiple comparisons among means. *Journal of the American Statistical Association, 56*(293), 52–64. https://doi.org/10.1080/01621459.1961.10482090

Feller, W. (1950). *An introduction to probability theory and its applications: Vol. One*. John Wiley & Sons.

Hochberg, Y. (1988). A sharper Bonferroni procedure for multiple tests of significance. *Biometrika, 75*(4), 800–802. doi:10.1093/biomet/75.4.800

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics, 6*(2), 65–70.

Hommel, G. (1988). A stagewise rejective multiple test procedure based on a modified Bonferroni test. *Biometrika, 75*(2), 383–386. doi:10.1093/biomet/75.2.383

SAS. (n.d.). PROC MULTTEST: p-Value Adjustments. SAS/STAT(R) 9.22 User's Guide. Retrieved February 1, 2025, from https://support.sas.com/documentation/cdl/en/statug/63347/HTML/default/viewer.htm#st

Šidák, Z. (1967). Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association, 62*(318), 626. doi:10.2307/2283989

Wright, S. P. (1992). Adjusted p-values for simultaneous inference. *Biometrics, 48*(4), 1005. doi:10.2307/2532694

---

`r_goodman_kruskal_gamma`

# *Goodman-Kruskal Gamma*

---

## Description

A rank correlation coefficient. It ranges from -1 (perfect negative association) to 1 (perfect positive association). A zero would indicate no correlation at all.

A positive correlation indicates that if someone scored high on the first field, they also likely score high on the second, while a negative correlation would indicate a high score on the first would give a low score on the second.

Alternatives for Gamma are Kendall Tau, Stuart-Kendall Tau and Somers D, but also Spearman rho could be considered.

Gamma looks at so-called discordant and concordant pairs, and ignores tied pairs. Kendall Tau b does the same, but applies a correction for ties. Stuart-Kendall Tau c also, but also takes the size of the table into consideration. Somers d only makes a correction for tied pairs in one of the two directions. Spearman rho is more of a variation on Pearson correlation, but applied to ranks. See Göktaş and İşçi. (2011) for more information on the comparisons.

### Usage

```
r_goodman_kruskal_gamma(
  ordField1,
  ordField2,
  levels1 = NULL,
  levels2 = NULL,
  ase = "appr",
  useRanks = FALSE
)
```

### Arguments

| | |
|---|---|
| `ordField1` | the numeric scores of the first variable |
| `ordField2` | the numeric scores of the second variable |
| `levels1` | vector, optional. the categories to use from ordField1 |
| `levels2` | vector, optional. the categories to use from ordField2 |
| `ase` | optional. Which asymptotic standard error to use. Either "appr" (default), 0, 1 |
| `useRanks` | boolean, optional. rank the data first or not. Default is False |

### Details

The formula used (Goodman & Kruskal, 1954, p. 749):

$$\gamma = \frac{P - Q}{P + Q}$$

With:

$$P = \sum_{i=1}^{r}\sum_{j=1}^{c} P_{i,j}$$

$$Q = \sum_{i=1}^{r}\sum_{j=1}^{c} Q_{i,j}$$

$$P_{i,j} = F_{i,j} \times C_{i,j}$$

$$Q_{i,j} = F_{i,j} \times D_{i,j}$$

$$C_{i,j} = \sum_{h<i}\sum_{k<j} F_{h,k} + \sum_{h>i}\sum_{k>j} F_{h,k}$$

$$D_{i,j} = \sum_{h<i}\sum_{k>j} F_{h,k} + \sum_{h>i}\sum_{k<j} F_{h,k}$$

The test can be done with a generic approximation:

$$z_{\gamma} = \gamma \times \sqrt{\frac{P + Q}{n \times \left(1 - \gamma^2\right)}}$$

If we assume the alternative hypothesis we can obtain (Goodman & Kruskal, 1963, p. 324; Goodman & Kruskal, 1972, p. 416; Brown & Benedetti, 1977, p. 310):

$$z_{\gamma} = \frac{\gamma}{ASE_1}$$

$$ASE_1 = \frac{4}{\left(P + Q\right)^2} \times \sqrt{\sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(Q \times C_{i,j} - P \times D_{i,j}\right)^2}$$

While if we assume the null hypothesis we can obtain (Brown & Benedetti, 1977, p. 311):

$$z_{\gamma} = \frac{\gamma}{ASE_0}$$

$$ASE_0 = \frac{2}{P + Q} \times \sqrt{\sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(C_{i,j} - D_{i,j}\right)^2 - \frac{\left(P - Q\right)^2}{n}}$$

The significance (p-value) in each case is then determined using:

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_{gamma}\right|\right)\right)$$

*Symbols*

- $F_{i,j}$ the count of scores equal to i in the first variable and j in the second
- $n$ the total sample size
- $r$ the number of unique categories in the first variable (number of rows)
- $c$ the number of unique categories in the second variable (number of columns)
- $P$ double the number of concordant pairs
- $Q$ double the number of discordant pairs

Note that Kendall $\tau_a$ is the same as Goodman-Kruskall gamma.

**Alternatives**

*library(DescTools)*

GoodmanKruskalGamma(table(ord1, ord2), conf.level=0.95)

*library(MESS)*

gkgamma(table(ord1, ord2))

*library(ryouready)*

ord.gamma(table(ord1, ord2))

## Value

A dataframe with:

| | |
|---|---|
| `g` | the Goodman-Kruskal Gamma value |
| `statistic` | the z-value used for the test |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Brown, M. B., & Benedetti, J. K. (1977). Sampling behavior of test for correlation in two-way contingency tables. *Journal of the American Statistical Association, 72*(358), 309–315. doi:10.2307/2286793

Göktaş, A., & İşçi, Ö. (2011). A comparison of the most commonly used measures of association for doubly ordered square contingency tables via simulation. *Advances in Methodology and Statistics, 8*(1). doi:10.51936/milh5641

Goodman, L. A., & Kruskal, W. H. (1954). Measures of association for cross classifications. J*ournal of the American Statistical Association, 49*(268), 732–764. doi:10.2307/2281536

Goodman, L. A., & Kruskal, W. H. (1963). Measures of Association for Cross Classifications III: Approximate Sampling Theory. *Journal of the American Statistical Association, 58*(302), 310–364. doi:10.1080/01621459.1963.10500850

Goodman, L. A., & Kruskal, W. H. (1972). Measures of Association for Cross Classifications IV: Simplification of Asymptotic Variances. *Journal of the American Statistical Association, 67*(338), 415–421. doi:10.1080/01621459.1972.10482401

---

## `r_kendall_tau` *Kendall Tau (a and b)*

### Description

A rank correlation coefficient. It ranges from -1 (perfect negative association) to 1 (perfect positive association). A zero would indicate no correlation at all.

A positive correlation indicates that if someone scored high on the first field, they also likely score high on the second, while a negative correlation would indicate a high score on the first would give a low score on the second.

Alternatives for Gamma are Kendall Tau, Stuart-Kendall Tau and Somers D, but also Spearman rho could be considered.

Kendall Tau b looks at so-called discordant and concordant pairs, but unlike Gamma it does not ignore tied pairs. Stuart-Kendall Tau c also, but also takes the size of the table into consideration. Somers d only makes a correction for tied pairs in one of the two directions. Spearman rho is more of a variation on Pearson correlation, but applied to ranks. See Göktaş and İşçi. (2011) for more information on the comparisons.

Kendall Tau a is the same as Goodman-Kruskal Gamma. See r_stuart_tau() for Stuart-Kendall-Tau c.

### Usage

```
r_kendall_tau(
  ordField1,
  ordField2,
  levels1 = NULL,
  levels2 = NULL,
  version = c("a", "b"),
  test = c("kendall-appr", "bb", "as71", "kendall-exact"),
  cc = FALSE,
  useRanks = FALSE
)
```

### Arguments

| | |
|---|---|
| `ordField1` | the numeric scores of the first variable |
| `ordField2` | the numeric scores of the second variable |
| `levels1` | vector, optional. the categories to use from ordField1 |
| `levels2` | vector, optional. the categories to use from ordField2 |
| `version` | string, optional. tau to be determined. Either "b" (default) or "a" |
| `test` | string, optional. Which test to use. Only applies if version="b". Either "bb" (default), "kendall-appr", "as71", "kendall-exact" |
| `cc` | boolean to indicate the use of a continuity correction |
| `useRanks` | boolean, optional. rank the data first or not. Default is False |

### Details

Kendall tau looks at concordant pairs versus discordant pairs. These can be calculated using:

$$n_c = \frac{P}{2}$$

$$n_d = \frac{Q}{2}$$

If the scores are placed in a cross table we can use:

$$P = \sum_{i=1}^{r}\sum_{j=1}^{c} P_{i,j}$$

$$Q = \sum_{i=1}^{r}\sum_{j=1}^{c} Q_{i,j}$$

$$P_{i,j} = F_{i,j} \times C_{i,j}$$

$$Q_{i,j} = F_{i,j} \times D_{i,j}$$

$$C_{i,j} = \sum_{h<i}\sum_{k<j} F_{h,k} + \sum_{h>i}\sum_{k>j} F_{h,k}$$

$$D_{i,j} = \sum_{h<i}\sum_{k>j} F_{h,k} + \sum_{h>i}\sum_{k<j} F_{h,k}$$

Alternative, we don't have to use a cross table:

$$P = \sum_{i=1}^{n} P_i$$

$$Q = \sum_{i=1}^{n} Q_i$$

$$P_i = \sum_{j=1}^{n} \begin{cases} 1 & \text{sign}\left(x_i - x_j\right) \times \text{sign}\left(y_i - y_j\right) = 1 \\ 0 & \text{else} \end{cases}$$

$$Q_i = \sum_{j=1}^{n} \begin{cases} 1 & \text{sign}\left(x_i - x_j\right) \times \text{sign}\left(y_i - y_j\right) = -1 \\ 0 & \text{else} \end{cases}$$

#' *Symbols used:*

- $n_c$ the number of concordant pairs
- $n_d$ the number of discondant pairs
- $n$ is the number of pairs
- $r$ the number of categories in the first variable (i.e. number of rows)
- $c$ the number of categories in the second variable (i.e. number of columns)
- $F_{h,k}$ is the number of cases that scored h for the first variable, and k for the second
- $P$ is double the number of concordant pairs
- $Q$ is double the number of discordant pairs

**Tau a**

The formula used for the value of $\tau_a$ is (Kendall, 1938, p. 82):

$$\tau_a = \frac{P - Q}{n \times (n-1)}$$

The formula can also be written as:

$$\tau_a = \frac{n_c - n_d}{n_0}$$

With:

$$n_0 = \frac{n \times (n-1)}{2}$$

Note that Kendall $\tau_a$ is the same as Goodman-Kruskall gamma.

**Tau b**

For $\tau_b$ the formula used is (Kendall, 1945, p. 243):

$$\tau_b = \frac{P - Q}{\sqrt{D_r \times D_c}}$$

With:

$$D_r = n^2 - \sum_{i=1}^{r} RS_i^2$$

$$D_c = n^2 - \sum_{j=1}^{c} CS_i^2$$

$$RS_i = \sum_{j=1}^{c} F_{i,j}$$

$$CS_i = \sum_{i=1}^{r} F_{i,j}$$

Note that $RS_i$ are the frequencies of the scores in the first variable and $CS_i$ are the frequencies of the scores in the second variable.

Alternative the formula can be written as:

$$\tau_b = \frac{n_c - n_d}{\sqrt{(n_0 - t_1) \times (n_0 - t_2)}}$$

With:

$$t_1 = \sum_{i=1}^{n} \frac{RS_i * (RS_i - 1)}{2}$$

$$t_2 = \sum_{i=1}^{n} \frac{CS_i * (CS_i - 1)}{2}$$

**Testing**

For *Tau a* the following normal approximation can be used (Kendall, 1962, p. 51):

$$z_a = \frac{3 \times \left(\frac{P-Q}{2}\right)}{\sqrt{\frac{n \times (n-1) \times (2 \times n+5)}{2}}}$$

$$sig. = 2 \times (1 - \Phi(|z_a|))$$

Or written with $\tau_a$ (Schaeffer & Levitt, p. 341):

$$z_a = \frac{\tau_a}{\sqrt{\frac{\sigma_{\tau_a}^2}{n}}}$$

With:

$$\sigma_{\tau_a}^2 = \frac{4 \times n + 10}{9 \times n \times (n-1)}$$

For *Tau b* an approximation can be used with:

$$z_b = \frac{\tau_b}{ASE}$$

For the equation of ASE two variations to choose from: Using Brown and Benedetti (1977, p. 311):

$$ASE_0 = 2 \times \sqrt{\frac{\sum_{i=1}^{r} \sum_{j=1}^{c} F_{i,j} \times (C_{i,j} - D_{i,j})^2 - \frac{(P-Q)^2}{n}}{D_r \times D_c}}$$

This is used when *test="bb"*

The calculation of $\sum_{j=1}^{c} F_{i,j} \times (C_{i,j} - D_{i,j})^2$ can then also be accomplished using:

$$\sum_{i=1}^{n} (P_i - Q_i)^2$$

Or a version from Kendall (1962, p. 55):

$$ASE = 2 \times \sqrt{v}$$

With:

$$v = \frac{v_0 - v_r - v_c}{18} + v_1 + v_2$$

$$v_0 = n \times (n-1) \times (2 \times n + 5)$$

$$v_r = \sum_{i=1}^{r} RS_i \times (RS_i - 1) \times (2 \times RS_i + 5)$$

$$v_c = \sum_{j=1}^{c} CS_j \times (CS_j - 1) \times (2 \times CS_j + 5)$$

$$v_1 = \frac{\left(\sum_{i=1}^{r} RS_i \times (RS_i - 1) \times (RS_i - 2)\right) \times \left(\sum_{j=1}^{c} CS_j \times (CS_j - 1) \times (CS_j - 2)\right)}{9 \times n \times (n-1) \times (n-2)}$$

This is used when *test="kendall"* and the default

For the AS 71 algorithm the test statistic is given by:

$$S = \binom{n}{2} \times |\tau| = \frac{n \times (n-1)}{2} \times |\tau|$$

The absolute value of $\tau$ is used, since AS 71 only provides the upper-tail probabilities.

For the Kendall method the test statistic is the number of concordant pairs $n_c$.

See the documentation of *di_kendall_tau()* for more info on the AS 71 and Kendall algorithms.

The continuity correction is applied as (Schaeffer & Levitt, p. 342):

$$\tau_{cc} = |\tau| - \frac{2}{n \times (n-1)}$$

or depending on the approximation used (Kendall, 1961, p. 54):

$$S_{cc} = |S| - 1$$

Where:

$$S = n_c - n_d$$

Note that this correction should actually be adjusted in case ties are present. Hopefully this can be implemented in a future update.

**Alternatives**

*R's stats library*

cor.test(ord1, ord2, method="kendall", exact=FALSE)

cor.test(ord1, ord2, method="kendall", exact=TRUE)

cor.test(ord1, ord2, method="kendall", exact=FALSE, continuity = TRUE)

*library(ryouready)*

ord.tau(table(ord1, ord2))

## Value

A dataframe with:

| | |
|---|---|
| `tau` | the tau value |
| `statistic` | the statistic from the test (z-value) |
| `pValue` | the significance (p-value) |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Best, D. J., & Gipps, P. G. (1974). Algorithm AS 71: The upper tail probabilities of Kendall's tau. *Applied Statistics, 23*(1), 98–100. doi:10.2307/2347062

Brown, M. B., & Benedetti, J. K. (1977). Sampling behavior of test for correlation in two-way contingency tables. *Journal of the American Statistical Association, 72*(358), 309–315. doi:10.2307/2286793

Göktaş, A., & İşçi, Ö. (2011). A comparison of the most commonly used measures of association for doubly ordered square contingency tables via simulation. *Advances in Methodology and Statistics, 8*(1). doi:10.51936/milh5641

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika, 30*(1–2), 81–93. doi:10.1093/biomet/30.1-2.81

Kendall, M. G. (1945). The treatment of ties in ranking problems. *Biometrika, 33*(3), 239–251. doi:10.1093/biomet/33.3.239

Kendall, M. G. (1962). *Rank correlation methods* (3rd ed.). Charles Griffin.

Kendall, M., & Gibbons, J. D. (1990). *Rank correlation methods* (5th ed.). Oxford University Press.

Schaeffer, M. S., & Levitt, E. E. (1956). Concerning Kendall's tau, a nonparametric correlation coefficient. *Psychological Bulletin, 53*(4), 338–346. doi:10.1037/h0045013

---

# `r_pearson` *Pearson Correlation Coefficient*

## Description

A measure of linear correlation. A -1 indicates a perfect negative linear correlation (i.e. a straight line going down, if the score in one field goes up, the other one goes down), a 0 would indicate no correlation, and a +1 a perfect positive linear correlation (i.e. a straight line going up, if the score in one field goes up, the other one goes up as well).

Various tests can be used to determine if the coefficient is significantly different from zero. See notes for details.

## Usage

```
r_pearson(
  field1,
  field2,
  corr = c("none", "wherry", "fisher", "olkin-pratt-1", "olkin-pratt-2",
    "olkin-pratt-3", "smith", "cattin", "pratt", "herzberg"),
  test = c("t", "z")
)
```

## Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `corr` | string, optional. Which adjustment to make if any (default is "none") |
| `test` | string, optional. Which test to use (see details). Either "t" (default), "z" |

## Details

This function makes use of the *hyperg_2F1* function from the *gsl* library.

The formula used (Pearson, 1896, p. 265):

$$r = \frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right) \times \left(y_i - \bar{y}\right)}{SS_x \times SS_y}$$

With:

$$SS_x = \sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2$$

$$SS_y = \sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

**Symbols used:**

- $n$ the number of pairs (sample size)
- $x_i$ the i-th score in the first variable
- $y_i$ the i-th score in the second variable

The test if test="t" is used is from Pugh and Winslow (1966, pp. 196,199):

$$sig. = 2 \times \left(1 - \text{T}\left(\left|t_r\right|, df\right)\right)$$

With:

$$t_r = r \times \sqrt{\frac{n-2}{1-r^2}}$$

$$df = n - 2$$

The test if test="z" is used is based on a Fisher transformation (Fisher, 1915, p. 521):

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_r\right|\right)\right)$$

With:

$$z_r = \text{atanh}\left(r\right) \times \sqrt{n-3}$$

This is derived since the Fisher transformation has a standard error of:

$$SE = \frac{1}{n-3}$$

As a source for this standard error Fisher (1921) is sometimes reported, but couldn't clearly find it in there. It can for example be found in Steiger (1980, p. 246) who refers to Olkin and Siotani (1964).

The correlation coefficient is biased and can be adjusted. There are many different adjustments suggested. For a great overview see Raju et al. (1997).

Fisher (1915, p. 521) - adj="fisher":

$$r_{adj} = r \times \left(1 + \frac{1-r^2}{2 \times n}\right)$$

Smith (Ezekiel, 1929, p. 100) - adj="smith":

$$r_{adj} = \sqrt{1 - \frac{1 - r^2}{1 - \frac{2}{n}}} = \sqrt{1 - \frac{n}{n - 2} \times (1 - r^2)}$$

Wherry (1931, p. 451) - adj="wherry":

$$r_{adj} = \sqrt{\frac{(n - 1) \times r^2 - 1}{n - 2}} = \sqrt{1 - (1 - r^2) \times \frac{n - 1}{n - 2}}$$

Ezekiel (1930 as cited in Raju et al., 1997, p. 295) - adj="ezekiel":

$$r_{adj} = \sqrt{1 - \frac{n - 1}{n - 3} \times (1 - r^2)}$$

Olkin-Pratt (1958, p. 211) - adj="olkin-pratt-1":

$$r_{adj} = r \times \text{HG}\left(\frac{1}{2}, \frac{1}{2}, \frac{n - 1}{2}, 1 - r^2\right)$$

Olkin-Pratt (1958, p. 203) - adj="olkin-pratt-2"

$$r_{adj} = r \times \left(1 + \frac{1 - r^2}{2 \times (n - 3)}\right)$$

Cattin (1980a, p. 64; 1980b, p. 409) - adj="cattin":

$$r_{adj} = \sqrt{1 - (1 - r^2) \times \left(1 + \frac{2 \times (1 - r^2)}{n - 1} + \frac{8 \times (1 - r^2)^2}{(n - 3) \times (n + 1)}\right)}$$

Pratt (1964, as cited in Claudy, 1978, p. 597) - adj="pratt":

$$r_{adj} = \sqrt{1 - (1 - r^2) \times \left(1 + \frac{2 \times (1 - r^2)}{n - 4.3}\right)}$$

Herzberg (1969, p. 5) - adj="herzberg":

$$r_{adj} = \sqrt{1 - (1 - r^2) \times \left(1 + \frac{2 \times (1 - r^2)}{n - 1}\right)}$$

Claudy (1978, p. 603) - adj="claudy":

$$r_{adj} = \sqrt{1 - \frac{n - 4}{n - 2} \times (1 - r^2) \times \left(1 + \frac{2 \times (1 - r^2)}{n - 1}\right)}$$

## Alternatives

R's *stats* library has a similar function.

cor.test(var1, var2)

## Value

A dataframe with:

| | |
|---|---|
| `r` | the Pearson Correlation Coefficient |
| `statistic` | the test statistic |
| `df` | degrees of freedom (only applicable for t test) |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cattin, P. (1980a). Note on the estimation of the squared cross-validated multiple correlation of a regression model. *Psychological Bulletin, 87*(1), 63–65. https://doi.org/10.1037/0033-2909.87.1.63

Cattin, P. (1980b). Estimation of the predictive power of a regression model. *Journal of Applied Psychology, 65*(4), 407–414. https://doi.org/10.1037/0021-9010.65.4.407

Claudy, J. G. (1978). Multiple regression and validity estimation in one sample. *Applied Psychological Measurement, 2*(4), 595–607. https://doi.org/10.1177/014662167800200414

Ezekiel, M. (1929). The application of the theory of error to multiple and curvilinear correlation. *Journal of the American Statistical Association, 24*(165), 99–104. https://doi.org/10.2307/2277015

Ezekiel, M. (1941). Methods of correlation analysis (2nd ed.). John Wiley & Sons.Fisher, R. A. (1915). Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population. *Biometrika, 10*(4), 507–521. https://doi.org/10.2307/2331838

Fisher, R. A. (1921). On the “probable error” of a coefficient of correlation deduced from a small sample. *Metron, 1*, 3–32.

Herzberg, P. A. (1969). The parameters of cross-validation. *Psychometrika Monograph Supplement, 34*(2), 1–70.

Olkin, I., & Pratt, J. W. (1958). Unbiased estimation of certain correlation coefficients. *The Annals of Mathematical Statistics, 29*(1), 201–211. https://doi.org/10.1214/aoms/1177706717

Olkin, I., & Siotani, M. (1964). Asymptotic distribution of functions of a correlation matrix. Laboratory for Quantitative Research in Education, Department of Statistics, Stanford University.

Pearson, K. (1896). Mathematical Contributions to the Theory of Evolution. III. Regression, Heredity, and Panmixia. *Philosophical Transactions of the Royal Society of London. *(A.), 1896, 253–318.

Pugh, E. M., & Winslow, G. H. (1966). *The analysis of physical measurements*. Addison-Wesley.

Raju, N. S., Bilgic, R., Edwards, J. E., & Fleer, P. F. (1997). Methodology review: Estimation of population validity and cross-validity, and the use of equal weights in prediction. *Applied Psychological Measurement, 21*(4), 291–305. https://doi.org/10.1177/01466216970214001

Steiger, J. H. (1980). Tests for comparing elements of a correlation matrix. *Psychological Bulletin, 87*(2), 245–251. https://doi.org/10.1037/0033-2909.87.2.245

Wherry, R. J. (1931). A new formula for predicting the shrinkage of the coefficient of multiple correlation. *The Annals of Mathematical Statistics, 2*(4), 440–457. https://doi.org/10.1214/aoms/1177732951

---

## `r_point_biserial` *Point Biserial Correlation Coefficient*

---

### Description

This can be seen as coding a binary variable with the groups into 0 and 1, and then calculates a Pearson correlation coefficient between the those values and the scores.

This gives the same result as the formula used and as input the Student t-test statistic and corresponding degrees of freedom.

### Usage

```
r_point_biserial(t, df)
```

### Arguments

| | |
|---|---|
| `t` | the test statistic value |
| `df` | the degrees of freedom |

### Details

The formula used is (Friedman, 1968, p. 245):

$$r_{pb} = \sqrt{\frac{t^2}{t^2 + df}}$$

*Symbols used:*

- $t$ the test statistic of the independent samples Student t-test
- $df$ the degrees of freedom of the independent samples Student t-test

### Value

Point Biserial Correlation Coefficient

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Friedman, H. (1968). Magnitude of experimental effect and a table for its rapid estimation. *Psychological Bulletin, 70*(4), 245–251. https://doi.org/10.1037/h0026258

### See Also

`ts_student_t_is`, Student t-test

### Examples

```
r_point_biserial(0.9984, 1967)
```

## `r_polychoric` *Polychoric Correlation Coefficient*

### Description

Polychoric Correlation Coefficient

### Usage

```
r_polychoric(dataVar, groupVar)
```

### Arguments

| | |
|---|---|
| `dataVar` | A vector with the scores data |
| `groupVar` | A vector with the group data |

### Details

This simply uses the *polychor()* function from the *polycor* library

### Value

Polychoric Correlation Coefficient value

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### Examples

```
scores = c(5, 12, 3, 4, 6, 1, 11, 13, NA)
groups = c("A","A","A","B","B","B","B", NA, "C")
r_polychoric(scores, groups)
```

## `r_rank_biserial_is` *(Glass) Rank Biserial Correlation / Cliff Delta*

### Description

This function will calculate Rank biserial correlation coefficient (independent-samples)

### Usage

```
r_rank_biserial_is(catField, ordField, categories = NULL, levels = NULL)
```

## Arguments

| | |
|---|---|
| `catField` | A vector with the scores data |
| `ordField` | A vector with the group data |
| `categories` | optional vector with categories to use and order for the categorical field. Otherwise the first two found will be used. |
| `levels` | optional vector with the labels of the ordinal field in order. |

## Details

The formula used is (Glass, 1966, p. 626):

$$r_b = \frac{2 \times \left(\bar{R}_1 - \bar{R}_2\right)}{n}$$

With:

$$\bar{R}_i = \frac{R_i}{n_i}$$

*Symbols used:*

- $\bar{R}_i$ the average of ranks in category i
- $R_i$ the sum of ranks in category i
- $n$ the total sample size
- $n_i$ the number of scores in category i

Glass (1966) showed that the formula was the same as that of the rank biserial from Cureton (1956). Cliff's delta (Cliff, 1993, p. 495) is actually also the same.

The rank biserial can be converted to a Cohen d (using the **es_convert()** function), and then the rules-of-thumb for Cohen d could be used (**th_cohen_d**())

## Value

(Glass) Rank Biserial Correlation / Cliff Delta value

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cliff, N. (1993). Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological Bulletin, 114*(3), 494–509. https://doi.org/10.1037/0033-2909.114.3.494

Cureton, E. E. (1956). Rank-biserial correlation. *Psychometrika, 21*(3), 287–290. https://doi.org/10.1007/BF02289138

Glass, G. V. (1966). Note on rank biserial correlation. *Educational and Psychological Measurement, 26*(3), 623–631. https://doi.org/10.1177/001316446602600307

## See Also

`es_convert`, to convert to Cohen d, use `fr="rb", to="cohend"`.

`th_cohen_d`, rules of thumb for Cohen d

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
myLevels = c('Not scientific at all', 'Not too scientific', 'Pretty scientific', 'Very scientific')
r_rank_biserial_is(df1['sex'], df1['accntsci'], levels=myLevels)

#Example 2: vectors
binary = c("apple", "apple", "apple", "peer", "peer", "peer", "peer")
ordinal = c(4, 3, 1, 6, 5, 7, 2)
r_rank_biserial_is(binary, ordinal, categories=c("peer", "apple"))
```

---

## `r_rank_biserial_os` *Rank biserial correlation coefficient (one-sample)*

### Description

This function will calculate Rank biserial correlation coefficient (one-sample)

### Usage

```
r_rank_biserial_os(data, levels = NULL, mu = NULL)
```

### Arguments

| | |
|---|---|
| `data` | vector with the numeric scores |
| `levels` | optional vector with levels in order |
| `mu` | optional parameter to set the hypothesized median. If not used the midrange is used |

### Details

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

The formula used (Kerby, 2014, p. 5):

$$r_{rb} = \frac{\left|R_{pos} - R_{neg}\right|}{R}$$

This is actually the same as (King & Minium, 2008, p. 403):

$$r_{rb} = \frac{4 \times \left|R_{min} - \frac{R_{pos}+R_{min}}{2}\right|}{n \times (n+1)}$$

*Symbols used:*

- $R_{pos}$ the sum of the ranks with a positive deviation from the hypothesized median
- $R_{neg}$ the sum of the ranks with a positive deviation from the hypothesized median
- $R_{min}$ the minimum of $R_{pos}$,$R_{neg}$
- $n$ the number of ranks with a non-zero difference with the hypothesized median

- $R$ the sum of all ranks, i.e. $R_{pos} + R_{neg}$

If no hypothesized median is provided, the midrange is used, defined as:

$$\frac{x_{max} - x_{min}}{2}$$

Where $x_{max}$ is the maximum value of the scores, and $x_{min}$ the minimum

## Value

dataframe with the hypothesized median (mu) and the effect size measure

## Before, After and Alternatives

Before this measure you might want to perform the test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

After this you might want a rule-of-thumb: `th_rank_biserial`, for Rank Biserial Correlation rule-of-thumb

Alternative effect size measure with ordinal data: `es_common_language_os`, for the Common Language Effect Size. `es_dominance`, for the Dominance score. `r_rosenthal`, for the Rank-Biserial Correlation

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Kerby, D. S. (2014). The simple difference formula: An approach to teaching nonparametric correlation. *Comprehensive Psychology*, 3, 1–9. https://doi.org/10.2466/11.IT.3.1

King, B. M., & Minium, E. W. (2008). *Statistical reasoning in the behavioral sciences* (5th ed.). John Wiley & Sons, Inc.

## Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
r_rank_biserial_os(ex1, levels=order)

ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
r_rank_biserial_os(ex2)
```

---

## r_rosenthal *Rosenthal Correlation Coefficient*

### Description

This function will calculate Rosenthal Correlation Coefficient. A simple correlation coefficient that divides a z-score by the square root of the sample size.

### Usage

```
r_rosenthal(zVal, n)
```

### Arguments

| | |
|---|---|
| `zVal` | z-value of test |
| `n` | total sample size |

### Details

This function is shown in this YouTube video and the effect size is also described at PeterStatistics.com

The formula used (Rosenthal, 1991, p. 19):

$$r = \frac{z}{\sqrt{n}}$$

*Symbols used:*

- $n$ the sample size
- $z$ the calculated z-statistic value

Rosenthal (1991) is the oldest reference I could find for this correlation coefficient. However, Cohen (1988, p. 275) actually has a measure 'f' that has the same equation.

For a classification the same as for Pearson correlation use *th_pearson_r()*

### Value

r the effect size measure

### Before, After and Alternatives

Before this measure you might want to perform the test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

After this you might want to use the rules-of-thumb for a Pearson Correlation or Cohen f: `th_pearson_r`, for rules of thumb for a Pearson correlation coefficient `th_cohen_f`, for rules of thumb for a Cohen f

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Rosenthal, R. (1991). *Meta-analytic procedures for social research* (Rev. ed). Sage Publications.

## Examples

```
z = 1.143943
n = 20
r_rosenthal(z, n)
```

---

# r_somers_d *Somers' d*

## Description

A rank correlation coefficient. It ranges from -1 (perfect negative association) to 1 (perfect positive association). A zero would indicate no correlation at all.

A positive correlation indicates that if someone scored high on the first field, they also likely score high on the second, while a negative correlation would indicate a high score on the first would give a low score on the second.

Alternatives for Somers D are Gamma, Kendall Tau, and Stuart-Kendall Tau, but also Spearman rho could be considered.

Kendall Tau b looks at so-called discordant and concordant pairs, but unlike Gamma it does not ignore tied pairs. Stuart-Kendall Tau c also, but also takes the size of the table into consideration. Somers d only makes a correction for tied pairs in one of the two directions. Spearman rho is more of a variation on Pearson correlation, but applied to ranks. See Göktaş and İşçi. (2011) for more information on the comparisons.

Kendall Tau a is the same as Goodman-Kruskal Gamma.

## Usage

```
r_somers_d(
  ordField1,
  ordField2,
  levels1 = NULL,
  levels2 = NULL,
  useRanks = FALSE
)
```

## Arguments

| | |
|---|---|
| `ordField1` | the numeric scores of the first variable |
| `ordField2` | the numeric scores of the second variable |
| `levels1` | vector, optional. the categories to use from ordField1 |
| `levels2` | vector, optional. the categories to use from ordField2 |
| `useRanks` | boolean, optional. rank the data first or not. Default is False |

## Details

**Asymmetric versions** The formula used is given by (Somers, 1962, p. 804):

$$d_{y|x} = \frac{P-Q}{D_r}, d_{x|y} = \frac{P-Q}{D_c}$$

With:

$$P = \sum_{i=1}^{r}\sum_{j=1}^{c} P_{i,j}$$

$$Q = \sum_{i=1}^{r}\sum_{j=1}^{c} Q_{i,j}$$

$$P_{i,j} = F_{i,j} \times C_{i,j}$$

$$Q_{i,j} = F_{i,j} \times D_{i,j}$$

$$C_{i,j} = \sum_{h<i}\sum_{k<j} F_{h,k} + \sum_{h>i}\sum_{k>j} F_{h,k}$$

$$D_{i,j} = \sum_{h<i}\sum_{k>j} F_{h,k} + \sum_{h>i}\sum_{k<j} F_{h,k}$$

$$D_r = n^2 - \sum_{i=1}^{r} RS_i^2$$

$$D_c = n^2 - \sum_{j=1}^{c} CS_i^2$$

$$RS_i = \sum_{j=1}^{c} F_{i,j}$$

$$CS_i = \sum_{i=1}^{r} F_{i,j}$$

*Symbols used:*

- $n$ is the number of pairs
- $r$ the number of categories in the first variable (i.e. number of rows)
- $c$ the number of categories in the second variable (i.e. number of columns)
- $F_{h,k}$ is the number of cases that scored h for the first variable, and k for the second
- $P$ is double the number of concordant pairs
- $Q$ is double the number of discordant pairs

Note that $RS_i$ are the frequencies of the scores in the first variable and $CS_i$ are the frequencies of the scores in the second variable.

For testing (SPSS, 2006, p. 121):

$$z_{y|x} = \frac{d_{y|x}}{ASE_{d_{y|x},0}}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_{y|x}\right|\right)\right)$$

With:

$$ASE_{d_{y|x},0} = \frac{2}{D_r} \times \sqrt{s}$$

$$s = \sum_{i} = 1^r \sum_{j=1}^{c} F_{i,j} \times \left(C_{i,j} - D_{i,j}\right)^2 - \frac{\left(P - Q\right)^2}{n}$$

and similar:

$$z_{x|y} = \frac{d_{x|y}}{ASE_{d_{x|y},0}}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_{x|y}\right|\right)\right)$$

With:

$$ASE_{d_{x|y},0} = \frac{2}{D_c} \times \sqrt{s}$$

**Symmetric**

The symmetric version is the same result as Kendall tau b and calculated using:

$$d = \frac{2 \times (P - Q)}{D_r + D_c}$$

and tested using:

$$z_d = \frac{d}{ASE_{d,0}}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_{y|x}\right|\right)\right)$$

With:

$$ASE_{d,0} = \frac{4}{D_r + D_c} \times \sqrt{s}$$

The function will also calculate the $ASE_1$ which are defined as:

$$ASE_{d_{y|x},1} = \frac{2 \times \sqrt{\sum_{i=1}^{3} \sum_{j=1}^{c} F_{i,j} \times \left(D_r \times \left(C_{i,j} - D_{i,j}\right) - (P - Q) \times \left(n - RS_i\right)\right)^2}}{D_r^2}$$

$$ASE_{d_{x|y},1} = \frac{2 \times \sqrt{\sum_{i=1}^{3} \sum_{j=1}^{c} F_{i,j} \times \left(D_c \times \left(C_{i,j} - D_{i,j}\right) - (P - Q) \times \left(n - CS_j\right)\right)^2}}{D_c^2}$$

$$ASE_{d,1} = \frac{2 \times ASE_{\tau_b,1}}{D_r + D_c} \times \sqrt{D_c \times D_c}$$

With:

$$ASE_{\tau_b,1} = \frac{\sqrt{\sum_{i=1}^{3} \sum_{j=1}^{c} F_{i,j} \times \left(2 \times \sqrt{D_r \times D_c} \times \left(C_{i,j} - D_{i,j}\right) + \tau_b \times v_{i,j}\right)^2 - n^3 \times \tau_b^2 \times \left(D_r + D_c\right)^2}}{D_r \times D_c}$$

$$v_{i,j} = RS_i \times D_c + CS_j \times D_r$$

**Alternatives**

*library(DescTools)*

SomersDelta(ord1, ord2, direction = "row")

SomersDelta(ord1, ord2, direction = "column")

*library(ryouready)*

ord.somers.d(table(ord1,ord2))

## Value

A dataframe with:

| | |
|---|---|
| `dependent` | which version (all three are in the rows) |
| `d` | the Sommers d value |
| `statistic` | the test statistic (z-value) |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Göktaş, A., & İşçi, Ö. (2011). A comparison of the most commonly used measures of association for doubly ordered square contingency tables via simulation. *Advances in Methodology and Statistics, 8*(1). doi:10.51936/milh5641

Somers, R. H. (1962). A new asymmetric measure of association for ordinal variables. *American Sociological Review, 27*(6), 799–811. doi:10.2307/2090408

SPSS. (2006). SPSS 15.0 algorithms.

---

# `r_spearman_rho` *Spearman Rho / Rank Correlation Coefficient*

## Description

The Spearman Rank Correlation Coefficient is the Pearson Correlation Coefficient, after the scores first have been converted to ranks.

This function makes use of *di_spearman()* for the test of this correlation, which requires the *pspearman* library for exact computations.

## Usage

```
r_spearman_rho(
  ordField1,
  ordField2,
  levels1 = NULL,
  levels2 = NULL,
  test = c("t", "z-fieller", "z-olds", "iman-conover", "as89", "exact"),
  cc = FALSE
)
```

## Arguments

| | |
|---|---|
| `ordField1` | the numeric scores of the first variable |
| `ordField2` | the numeric scores of the second variable |
| `levels1` | vector, optional. the categories to use from ordField1 |
| `levels2` | vector, optional. the categories to use from ordField2 |
| `test` | the test to be used. Either "t" (default), "as89", "exact", "iman-conover", "z-fieller", "z-olds", "none" |
| `cc` | boolean to indicate the use of a continuity correction |

## Details

The formula used is (Spearman, 1904, p. 77):

$$r_s = \frac{SS_{r_x,r_y}}{SS_{r_x}\times SS_{r_y}}$$

With:

$$SS_{r_x} = \sum_{i=1}^{n}\left(r_{x_i} - \bar{r}_x\right)^2$$

$$SS_{r_y} = \sum_{i=1}^{n}\left(r_{y_i} - \bar{r}_y\right)^2$$

$$SS_{r_x,r_y} = \sum_{i=1}^{n}\left(r_{x_i} - \bar{r}_x\right)\times\left(r_{y_i} - \bar{r}_y\right)$$

*Symbols*

- $r_{x_i}$ the i-th rank of the scores of the first variable
- $r_{y_i}$ the i-th rank of the scores of the second variable
- $n$ the total sample size (number of ranks)

If all the ranks are distinct (i.e. no ties) the formula can also be written as:

$$r_s = 1 - \frac{6}{n\times\left(n^2-1\right)}\times S$$

With:

$$S = \sum_{i=1}^{n} d_i^2$$

$$d_i^2 = \left(r_{x_i} - r_{y_i}\right)^2$$

The test can be performed in different ways. Options to choose from are:

- "t" uses a Student t distribution approximation
- "z-fieller" uses a standard normal approximation from Fieller
- "z-old" uses standard normal approximation from Old
- "iman-conover" a combination of z and t distribution from Iman and Conover
- "AS89" uses the AS 89 algorithm
- "exact" uses an exact distribution

See for the details of each the *di_spearman()* function

A continuity correction can be applied (Zar, 1972, p. 579):

$$r_s^{cc} = \left|r_s\right| - \frac{6}{n^3-n}$$

**Alternatives**

*R's stats*

Using the t-approximation:

cor.test(ord1, ord2, method="spearman")

Using AS89

cor.test(ord1, ord2, method="spearman", exact=TRUE)

*library(pspearman)*

spearman.test(ord1, ord2, approximation="t-distribution")

spearman.test(ord1, ord2, approximation="AS89")

spearman.test(ord1, ord2, approximation="exact")

## Value

A dataframe with:

| | |
|---|---|
| `rs` | the correlation coefficient |
| `pValue` | the significance (p-value) |
| `statistic` | the statistic from the test (only if applicable) |
| `df` | the degrees of freedom (only if applicable) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Göktaş, A., & İşçi, Ö. (2011). A comparison of the most commonly used measures of association for doubly ordered square contingency tables via simulation. *Advances in Methodology and Statistics, 8*(1). doi:10.51936/milh5641

Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology, 15*(1), 72–101.

Zar, J. H. (1972). Significance testing of the Spearman rank correlation coefficient. *Journal of the American Statistical Association, 67*(339), 578–580. doi:10.1080/01621459.1972.10481251

---

# `r_stuart_tau` *Stuart Tau c / Kendall Tau c*

## Description

A rank correlation coefficient. It ranges from -1 (perfect negative association) to 1 (perfect positive association). A zero would indicate no correlation at all.

A positive correlation indicates that if someone scored high on the first field, they also likely score high on the second, while a negative correlation would indicate a high score on the first would give a low score on the second.

Alternatives for Gamma are Kendall Tau, Stuart-Kendall Tau and Somers D, but also Spearman rho could be considered.

Kendall Tau b looks at so-called discordant and concordant pairs, but unlike Gamma it does not ignore tied pairs. Stuart-Kendall Tau c also, but also takes the size of the table into consideration. Somers d only makes a correction for tied pairs in one of the two directions. Spearman rho is more of a variation on Pearson correlation, but applied to ranks. See Göktaş and İşçi. (2011) for more information on the comparisons.

Kendall Tau a is the same as Goodman-Kruskal Gamma.

## Usage

```
r_stuart_tau(
  ordField1,
  ordField2,
  levels1 = NULL,
  levels2 = NULL,
  cc = FALSE,
  useRanks = FALSE
)
```

## Arguments

| | |
|---|---|
| `ordField1` | the numeric scores of the first variable |
| `ordField2` | the numeric scores of the second variable |
| `levels1` | vector, optional. the categories to use from ordField1 |
| `levels2` | vector, optional. the categories to use from ordField2 |
| `cc` | boolean to indicate the use of a continuity correction |
| `useRanks` | boolean, optional. rank the data first or not. Default is False |

## Details

Tau looks at concordant pairs versus discordant pairs. These can be calculated using:

$$n_c = \frac{P}{2}$$

$$n_d = \frac{Q}{2}$$

If the scores are placed in a cross table we can use:

$$P = \sum_{i=1}^{r}\sum_{j=1}^{c} P_{i,j}$$

$$Q = \sum_{i=1}^{r}\sum_{j=1}^{c} Q_{i,j}$$

$$P_{i,j} = F_{i,j} \times C_{i,j}$$

$$Q_{i,j} = F_{i,j} \times D_{i,j}$$

$$C_{i,j} = \sum_{h<i}\sum_{k<j} F_{h,k} + \sum_{h>i}\sum_{k>j} F_{h,k}$$

$$D_{i,j} = \sum_{h<i}\sum_{k>j} F_{h,k} + \sum_{h>i}\sum_{k<j} F_{h,k}$$

Alternative, we don't have to use a cross table:

$$P = \sum_{i=1}^{n} P_i$$

$$Q = \sum_{i=1}^{n} Q_i$$

$$P_i = \sum_{j=1}^{n} \begin{cases} 1 & \text{sign}\left(x_i - x_j\right) \times \text{sign}\left(y_i - y_j\right) = 1 \\ 0 & \text{else} \end{cases}$$

$$Q_i = \sum_{j=1}^{n} \begin{cases} 1 & \text{sign}\left(x_i - x_j\right) \times \text{sign}\left(y_i - y_j\right) = -1 \\ 0 & \text{else} \end{cases}$$

*Symbols used:*

- $n_c$ the number of concordant pairs
- $n_d$ the number of discondant pairs
- $n$ is the number of pairs
- $r$ the number of categories in the first variable (i.e. number of rows)
- $c$ the number of categories in the second variable (i.e. number of columns)
- $F_{h,k}$ is the number of cases that scored h for the first variable, and k for the second
- $P$ is double the number of concordant pairs
- $Q$ is double the number of discordant pairs

The formula used is (Stuart, 1953, p. 107):

$$\tau_c = \frac{P - Q}{n^2 \times \frac{m-1}{m}}$$

With:

$$m = \text{min}\left(r, c\right)$$

**Testing**

The following normal approximation can be used (Brown & Benedetti, 1977, p. 311):

$$z_c = \frac{\tau_c}{ASE}$$

$$ASE_0 = \frac{2 \times m}{\left(m - 1\right)^2} \times \sqrt{\sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} \times \left(C_{i,j} - D_{i,j}\right)^2 - \frac{\left(P - Q\right)^2}{n}}$$

The calculation of $\sum_{j=1}^{c} F_{i,j} \times \left(C_{i,j} - D_{i,j}\right)^2$ can then also be accomplished using:

$$\sum_{i=1}^{n} \left(P_i - Q_i\right)^2$$

The continuity correction is applied as (Schaeffer & Levitt, p. 342):

$$\tau_{cc} = \left|\tau\right| - \frac{2}{n \times \left(n - 1\right)}$$

Note that this correction should actually be adjusted in case ties are present. Hopefully this can be implemented in a future update.

**Alternatives**

*library(DescTools)*

StuartTauC(ord1, ord2)

*library(ryouready)*

ord.tau(table(ord1, ord2))

## Value

A dataframe with:

| | |
|---|---|
| `tau` | the tau value |
| `statistic` | the test statistic (z-value) |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Brown, M. B., & Benedetti, J. K. (1977). Sampling behavior of test for correlation in two-way contingency tables. *Journal of the American Statistical Association, 72*(358), 309–315. doi:10.2307/2286793

Göktaş, A., & İşçi, Ö. (2011). A comparison of the most commonly used measures of association for doubly ordered square contingency tables via simulation. *Advances in Methodology and Statistics, 8*(1). doi:10.51936/milh5641

Schaeffer, M. S., & Levitt, E. E. (1956). Concerning Kendall's tau, a nonparametric correlation coefficient. *Psychological Bulletin, 53*(4), 338–346. doi:10.1037/h0045013

Stuart, A. (1953). The estimation and comparison of strengths of association in contingency tables. *Biometrika, 40*(1/2), 105. doi:10.2307/2333101

---

# `r_tetrachoric` *Tetrachoric Correlation Coefficient*

## Description

In essence this attempts to mimic a correlation coefficient between two scale variables. It can be defined as "An estimate of the correlation between two random variables having a bivariate normal distribution, obtained from the information from a double dichotomy of their bivariate distribution" (Everitt, 2004, p. 372).

This assumes the two binary variables have 'hidden' underlying normal distribution. If so, the combination of the two forms a bivariate normal distribution with a specific correlation between them. The quest is then to find the correlation, such that the cumulative density function of the z-values of the two marginal totals of the top-left cell (a) match that value.

This is quite tricky to do, so a few have proposed an approximation for this. These include Yule r, Pearson Q4 and Q5, Camp, Becker and Clogg, and Bonett and Price , all available and more with (`es_bin_bin`).

Besides closed form approximation formula's, various algorithms have been designed as well. The three most often mentioned are Brown (1977), Kirk (1973), and Divgi (1979), available in this function.

## Usage

```
r_tetrachoric(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  method = "divgi"
)
```

## Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with selection and/or order for categories of field1 |
| `categories2` | : optional list with selection and/or order for categories of field2 |
| `method` | method to use (see details). Either "divgi" (default), "search", "kirk", "brown" |

## Details

The "search" method does a binary search for rt using the bivariate normal distribution from the *fMultivar* library.

"kirk" will use Kirk (1973) Fortran TET8 procedure, adapted by stikpet

"brown" will use Brown (1977) - Algorithm AS 116

"divgi" will use Divgi (1979) algorithm

Flow charts of these algorithms can be found at https://peterstatistics.com

## Value

Tetrachoric Correlation Coefficient

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Brown, M. B. (1977). Algorithm AS 116: The tetrachoric correlation and its asymptotic standard error. *Applied Statistics, 26*(3), 343. https://doi.org/10.2307/2346985

Divgi, D. R. (1979). Calculation of the tetrachoric correlation coefficient. *Psychometrika, 44*(2), 169–172. https://doi.org/10.1007/BF02293968

Kirk, D. B. (1973). On the numerical approximation of the bivariate normal (tetrachoric) correlation coefficient. *Psychometrika, 38*(2), 259–268. https://doi.org/10.1007/BF02291118

## Examples

```
#Example: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
r_tetrachoric(df1[['mar1']], df1[['sex']], categories1=c("WIDOWED", "DIVORCED"))
```

---

`srf` *Wilcoxon Sum-of-Ranks Frequency Function*

---

## Description

This helper function will give the count for a sum of ranks of T, given a sample size of n, using the recursive formula.

## Usage

```
srf(T, n)
```

## Arguments

| | |
|---|---|
| `T` | int with the sum of ranks |
| `n` | int with the sample size |

## Details

The recursive method uses the formula from McCornack (1965, p. 864):

$$srf\left(x, y\right) = \begin{cases} 0 & x < 0 \\ 0 & x > \binom{y+1}{2} \\ 1 & y = 1 \land \left(x = 0 \lor x = 1\right) \\ srf^*\left(x, y\right) & y \geq 0 \end{cases}$$

with:

$$srf^*\left(x, y\right) = srf\left(x - y, y - 1\right) + srf\left(x, y - 1\right)$$

## Value

A the requested count

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

McCornack, R. L. (1965). Extended tables of the Wilcoxon matched pair signed rank statistic. *Journal of the American Statistical Association, 60*(311), 864–871. doi:10.2307/2283253

---

`tab_cross` *Cross Table / Contingency Table*

---

### Description

A contingency table can be defined as “tables arising when observations on a number of categorical variables are cross-classified” (Everitt, 2004, p.89).

There are quite a few variations on the name for this type of table. Perhaps the oldest name is actually contingency table, which was the name Pearson (1904, p. 34) gave to them. Another popular name is cross tabulation (Upton & Cook, 2002, p. 79), but also cross classification table (Zekeck, 2014, p. 71) and bivariate frequency table (Porkess, 1988, p. 48) are used. The one I used cross table which can for example be found in Newbold et al. (2013, p. 9) or Sá (2007, p. 52).

### Usage

```
tab_cross(
  field1,
  field2,
  order1 = NULL,
  order2 = NULL,
  percent = c(NULL, "all", "row", "column"),
  totals = "exclude"
)
```

### Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `order1` | : optional list with order for categories of field1 |
| `order2` | : optional list with order for categories of field2 |
| `percent` | : optional which percentages to show. Either "none" (default), "all", "row", "column" |
| `totals` | : optional to add margin totals. Either "exclude" (default), or "include" |

### Value

dataframe : the cross table

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Everitt, B. (2004). *The Cambridge dictionary of statistics* (2nd ed.). Cambridge University Press. Newbold, P., Carlson, W. L., & Thorne, B. (2013). *Statistics for business and economics* (8th ed). Pearson.

Pearson, K. (1904). *Contributions to the Mathematical Theory of Evolution*. XIII. On the theory of contingency and its relation to association and normal correlation. Dulau and Co.

Porkess, R. (1988). *Dictionary of statistics*. Collins.

Sá, J. P. M. de. (2007). *Applied statistics: Using SPSS, Statistica, MATLAB, and R* (2nd ed.). Springer.

Upton, G., & Cook, I. (2002). *Oxford: Dictionary of statistics*. Oxford University Press.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

## Examples

```
#Example: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
tab_cross(df1[['mar1']], df1[['sex']], percent="column", totals="include")

orderR = c("DIVORCED", "WIDOWED", "SEPARATED", "MARRIED", "NEVER MARRIED")
orderC = c("MALE", "FEMALE")
tab_cross(df1[['mar1']], df1[['sex']], order1=orderR, order2=orderC)

order = c("Not scientific at all", "Not too scientific", "Pretty scientific", "Very scientific")
tab_cross(df1[['mar1']], df1[['accntsci']], order2=order)
```

---

# `tab_frequency` *Frequency Table*

## Description

A frequency table is defined as "a table showing (1) all of the values for a variable in a dataset, and (2) the frequency of each of those responses. Some frequency tables also show a cumulative frequency and proportions of responses" (Warne, 2017, p. 512).

A frequency table can help to get impression of your survey data of a binary, nominal, or ordinal variable. It could also help with a scale variable, provided there are not too many options. If, for example, you have asked for age, a list going from 1 to 90 with different ages and frequencies, will probably not be so helpful.

If you have many options in the scale variable, the data is often binned (e.g. 0 < 10, 10 < 20, etc.), which creates then an ordinal variable, of which a frequency table can then be helpful. See binning for more information on this.

A frequency table can show different types of frequencies. Various options are discussed in the details.

A YouTube video with explanation on this test is available here

## Usage

```
tab_frequency(data, order = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `order` | optional list with order of the categories |

## Details

The column **Frequency** shows how many respondents answered each option. We can tell that 100 people in this survey chose the option ’very scientific’. This is also known as the **absolute frequency** and defined as “the number of occurrences of a particular phenomenon” (Zedeck, “Frequency”, 2014, p. 144).

The **Percent** column shows the percentages, based on the grand total, so including the missing values. Percentages can be defined as “a way of expressing ratios in terms of whole numbers. A ratio or fraction is converted to a percentage by multiplying by 100 and appending a "percentage sign" %” (Weisstein, 2002, p. 2200).

The **Valid Percent** shows the percentage, based on the valid total, so excluding the missing values. Most often the ‘Percent’ shown in reports are actually Valid Percent, but the word ‘Valid’ is then simply left out.

Percentages show the number of cases that could be expected if there would be 100 cases in total, hence per-cent which means ’per 100’. If your sample size is very small, be careful about using percentages. If it is less than 100, it means that you are ’blowing up’ your differences, while percentages are more commonly used to ’scale down’.

The term **relative frequency** is also sometimes used. This is the frequency divided by the total number of cases. Note that this should then always produce a decimal value between 0 and 1 (inclusive). Multiply this by 100 and you get the percentage, multiply it by 1000 and you get permille (‰), multiply it by 360 and you get the degrees of a circle, etc.

In general the formula for a percentage is:

$$PR_i = \frac{F_i}{n} \times 100$$

*Symbols used:*

- $PR_i$ the percentage of category i
- $F_i$ the (absolute) frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies (either including or excluding the missing values)

The **cumulative frequency** (not shown in table) can be defined as: “the total (absolute) frequency up to the upper boundary of that class” (Kenney, 1939, p. 16). This would only be useful if there is an order to the categories, so we can say that for example 299 respondents found accounting pretty scientific or even more. Which is why these cumulative frequencies will not have a meaningful interpretation for a nominal variable (e.g. 28 students study business or less?).

The **Cumulative Percent** is the running total of the Valid Percent, it is the addition of all previous and the current category’s valid percentages.

The cumulative frequency can be calculated using:

$$CF_i = \sum_{j=1}^{i} F_j$$

Or using recursion:

$$CF_i = F_i + CF_{i-1}$$

For the cumulative percent the same formulas as for cumulative frequency can be used, but replacing $F_i$ with $PR_i$. It can also be determined using the cumulative frequency:

$$CPR_i = \frac{CF_i}{n}$$

When the categories are ranges of values (bins), the frequency density could become helpful. It can be defined as: “the number of occurrences of an event divided by the bin size...” (Zedeck, 2014, pp. 144–145). See the binned tables for more information about this.

## Value

Dataframe with the folowing columns:

| | |
|---|---|
| `index` | the categories |
| `frequency` | the absolute count |
| `percent` | the percentage based on the total including missing values |
| `valid percent` | the percentage based on the total excluding missing values, only if missing values are present |
| `cumulative percent` | the cumulative percentages |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Kenney, J. F. (1939). *Mathematics of statistics; Part one*. Chapman & Hall.

Warne, R. T. (2017). *Statistics for the social sciences: A general linear model approach*. Cambridge University Press.

Weisstein, E. W. (2002). *CRC concise encyclopedia of mathematics* (2nd ed.). Chapman & Hall/CRC.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
tab_frequency(ex1)

#Example 2: Text data with specified order
myOrder = c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED", "WIDOWED")
tab_frequency(df1['mar1'], order=myOrder)

#Example 3: Numeric data
ex3 = c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5)
tab_frequency(ex3)

#Example 4: Ordinal data
ex4a = c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, "NaN")
myOrder = c("fully disagree"=1, "disagree"=2, "neutral"=3, "agree"=4, "fully agree"=5)
tab_frequency(ex4a, order=myOrder)

ex4b = df1['accntsci']
myOrder = c("Not scientific at all", "Not too scientific", "Pretty scientific", "Very scientific")
tab_frequency(ex4b, order=myOrder)
```

## `tab_frequency_bins` *Binned Frequency Table*

### Description

Bins data and creates a frequency table with frequency density.

This function is shown in this YouTube video and frequency tables are also described at PeterStatistics.com

### Usage

```
tab_frequency_bins(
  data,
  nbins = "sturges",
  bins = NULL,
  incl_lower = TRUE,
  adjust = 1
)
```

### Arguments

| | |
|---|---|
| `data` | list or dataframe |
| `nbins` | optional, either the number of bins to create, or a specific method from the *tab_nbins()* function. Default is "sturges" |
| `bins` | optional dataframe with lower and upper bounds |
| `incl_lower` | optional boolean, to include the lower bound, otherwise the upper bound is included. Default is True |
| `adjust` | optional value to add or subtract to guarantee all scores will fit in a bin |

### Value

dataframe with:

| | |
|---|---|
| `lower bound` | lower bound of class |
| `upper bound` | upper bound of class |
| `frequency` | count of scores in bin |
| `frequency density` | count divided by bin range |

### Before, After and Alternatives

Before this you might want to determine the number of bins you use: `tab_nbins`, to determine the number of bins.

After this you might want to visualise the result: `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Numeric Dataframe
ex1a = df2['Gen_Age']
tab_frequency_bins(ex1a)

ex1b = df2['Gen_Age']
myBins = data.frame(c(0, 20, 25, 30), c(20, 25, 30, 120))
tab_frequency_bins(ex1b, bins=myBins)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
tab_frequency_bins(ex2, adjust=0.1)
```

---

# `tab_nbins` *Number of Bins*

## Description

To decide on the appropriate number of bins, many different rules can be applied. This function will determine the number of bins, based on the chosen method.

This function is shown in this YouTube video and binning is described at PeterStatistics.com

## Usage

```
tab_nbins(data, method = "src", adjust = 1, maxBins = NULL, qmethod = "cdf")
```

## Arguments

| | |
|---|---|
| `data` | vector or dataframe |
| `method` | optional to indicate the method to use. Either "src", "sturges", "qr", "rice", "ts", "exp", "velleman", "doane", "scott", "fd", "shinshim", "stone", or "knuth" |
| `adjust` | optional adjustment to upper bound to guarantee all scores will fit in range. |
| `maxBins` | optional for in iterations with "shinshim", "stone" and "knuth" |
| `qmethod` | optional quartile method calculation to use for IQR when "fd" is used. See me_quartiles for options |

## Details

The first few methods are determining the number of bins (k) using the sample size (n).

**Square Root Choice (src)**

This method uses (unknown source):

$$k = \lceil \sqrt{n} \rceil$$

**Sturges Choice (sturges)**

This method uses (Sturges, 1926, p. 65):

$$k = \lceil \log_2 (n) \rceil + 1$$

**Quartic Root (qr)**

This method uses (anonymous, as cited in Lohaka, 2007, p. 87):

$$k = \lceil 2.5 \times \sqrt[4]{n} \rceil$$

**Rice Rule (rice)**

This method uses (Lane, n.d., p. 85):

$$k = \lceil 2 \times \sqrt[3]{n} \lceil$$

**Terrell and Scotte (ts)**

This method uses (Terrell & Scott, 1985, p. 212):

$$k = \lceil \sqrt[3]{2 \times n} \rceil$$

**Exponential (exp)**

This method uses (Iman & Conover, 1989, p. 54):

$$k = \lceil \log_2 (n) \rceil$$

**Velleman (velleman)**

This method uses (Velleman, 1976 as cited in Lohaka, 2007, p. 89):

$$k = \begin{cases} \lceil 2 \times \sqrt{n} \rceil & \text{if } n \leq 100 \\ \lceil 10 \times \log_{10} (n) \rceil & \text{if } n > 100 \end{cases}$$

**Doane (doane)**

This method uses (Doane, 1976, pp. 181-182):

$$k = 1 + \lceil \log_2 (n) + \log_2 \left( 1 + \frac{|g_1|}{\sigma_{g_1}} \right) \rceil$$

In the formula's $g_1$ is the 3rd moment skewness:

$$g_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^3}{n \times \sigma^3} = \frac{1}{n} \times \sum_{i=1}^{n} \left( \frac{x_i - \bar{x}}{\sigma} \right)^3$$

With:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}}$$

The $\sigma_{g_1}$ is defined using the formula:

$$\sigma_{g_1} = \sqrt{\frac{6 \times (n-2)}{(n+1)(n+3)}}$$

Next are methods that determine the bin sizes (h), which can then be used to determine the number of bins (k) using:

$$k = \lceil \frac{\max (x) - \min (x)}{h} \rceil$$

**Scott (scott)**

This method uses (Scott, 1979, p. 608):

$$h = \frac{3.49 \times s}{\sqrt[3]{n}}$$

Where $s$ is the sample standard deviation:

$$s = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n-1}}$$

**Freedman and Diaconis (fd)**

This method uses (Freedman & Diaconis, 1981, p. 3):

$$h = 2 \times \frac{\text{IQR}\left(x\right)}{\sqrt[3]{n}}$$

Where IQR is the inter-quartile range.

The last three methods all minimize a cost function (or maximize a profit function). They make use of the following steps:

1. Divide the data into k bins and count the frequency in each bin
2. Compute the cost function
3. Repeat the first two steps while changing k, until a k is found that minimizes the cost function

**Shimazaki and Shinomoto (shinshim)**

This method uses as a cost function (Shimazaki & Shinomoto, 2007, p. 1508):

$$C_k = \frac{2 \times \bar{f}_k - \sigma_{f_k}}{h^2}$$

With $\bar{f}_k$ being the average of the frequencies when using k bins, and $\sigma_{f_k}$ the population variance. In formula notation:

$$\bar{f}_k = \frac{\sum_{i=1}^{k} f_{i,k}}{k}$$

$$\sigma_{f_k} = \frac{\sum_{i=1}^{k}\left(f_{i,k} - \bar{f}_k\right)^2}{k}$$

Where $f_{i,k}$ is the frequency of the i-th bin when using k bins.

**Stone (stone)**

This method uses as a cost function (Stone, 1984, p. 3):

$$C_k = \frac{1}{h} \times \left(\frac{2}{n-1} - \frac{n+1}{n-1} \times \sum_{i=1}^{k}\left(\frac{f_i}{n}\right)^2\right)$$

**Knuth (knuth)**

This method uses as a profit function (Knuth, 2019, p. 8):

$$P_k = n \times \ln\left(k\right) + \ln\Gamma\left(\frac{k}{2}\right) - k \times \ln\Gamma\left(\frac{1}{2}\right) - \ln\Gamma\left(n + \frac{k}{2}\right) + \sum_{i=1}^{k}\ln\Gamma\left(f_i + \frac{1}{2}\right)$$

## Value

k : integer with optimum number of bins according to chosen method

## Before, After and Alternatives

After this you might want to create a binned frequency table: `tab_frequency_bins`, to create a binned frequency table.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Doane, D. P. (1976). Aesthetic frequency classifications. *The American Statistician, 30*(4), 181–183. https://doi.org/10.2307/2683757

Freedman, D., & Diaconis, P. (1981). On the histogram as a density estimator. *Zeitschrift Für Wahrscheinlichkeitstheorie Und Verwandte Gebiete, 57*(4), 453–476. https://doi.org/10.1007/BF01025868

Iman, R. L., & Conover, W. J. (1989). *Modern business statistics* (2nd ed.). Wiley.

Knuth, K. H. (2019). Optimal data-based binning for histograms and histogram-based probability density models. *Digital Signal Processing, 95*, 1–30. https://doi.org/10.1016/j.dsp.2019.102581

Lohaka, H. O. (2007). Making a grouped-data frequency table: Development and examination of the iteration algorithm [Doctoral dissertation, Ohio University]. https://etd.ohiolink.edu

Scott, D. W. (1979). On optimal and data-based histograms. *Biometrika, 66*(3), 605–610. https://doi.org/10.1093/biomet/

Shimazaki, H., & Shinomoto, S. (2007). A method for selecting the bin size of a time histogram. *Neural Computation, 19*(6), 1503–1527. https://doi.org/10.1162/neco.2007.19.6.1503

Stone, C. J. (1984). An asymptotically optimal window selection rule for kernel density estimates. *The Annals of Statistics, 12*(4), 1285–1297.

Sturges, H. A. (1926). The choice of a class interval. *Journal of the American Statistical Association, 21*(153), 65–66. https://doi.org/10.1080/01621459.1926.10502161

Terrell, G. R., & Scott, D. W. (1985). Oversmoothed nonparametric density estimates. *Journal of the American Statistical Association, 80*(389), 209–214. https://doi.org/10.2307/2288074

## Examples

```
#Example 1
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['age']
ex1 = replace(ex1, ex1=="89 OR OLDER", "90")
tab_nbins(ex1)

#Example 2
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex2 = df2['Gen_Age']
tab_nbins(ex2)
```

---

`th_cle` *Rule-of-Thumb for Common Language Effect Size*

---

## Description

This function will give a qualification (classification) for a Common Language Effect Size (/ Vargha-Delaney A / Probability of Superiority)

The measure is also described at PeterStatistics.com

## Usage

```
th_cle(cle, qual = "vd", convert = "no")
```

## Arguments

| | |
|---|---|
| `cle` | the Vargha-Delaney A value |
| `qual` | "vd", others via convert, optional rules-of-thumb to use, currently only 'vd' for Vargha-Delaney, otherwise a converted measure |
| `convert` | "no", "rb", "cohen_d", optional list in case to use a rule-of-thumb from a converted measure. Either "no" for no conversion, "rb" for rank-biserial, or "cohen_d" for Cohen d. |

## Details

Vargha and Delaney (2000, p. 106):

| \|0.5 - A\| | Interpretation |
|---|---|
| 0.00 < 0.06 | negligible |
| 0.11 < 0.14 | small |
| 0.28 < 0.21 | medium |
| 0.21 or more | large |

The CLE can be converted to a Rank Biserial Coefficient using:

$$r_b = 2 \times CLE - 1$$

Rules of thumb from the **th_rank_biserial()** function could then be used, by setting: *convert="rb"*, and *qual* is any of the options in th_rank_biserial()

This in turn can be converted to Cohen's d using (Marfo & Okyere, 2019, p.4):

$$d = 2 \times \phi^{-1}\left(-\frac{1}{r_b - 2}\right)$$

Rules of thumb from the **th_cohen_d()** function could then be used, by setting: *convert="cohen_d"*, and *qual* is any of the options in th_cohen_d()

## Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

### Before, After and Alternatives

Before this you might want to obtain the measure: `es_common_language_os`, o determine the CLE for one-sample. `es_common_language_is`, o determine the CLE for independent samples.

The function uses the convert function and corresponding rules of thumb: `es_convert`, for the conversions. `th_rank_biserial`, for options for rules of thumb when converting to Rank Biserial. `th_cohen_d`, for options for rules of thumb when converting to Cohen d.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Marfo, P., & Okyere, G. A. (2019). The accuracy of effect-size estimates under normals and contaminated normals in meta-analysis. *Heliyon, 5*(6), e01838. doi:10.1016/j.heliyon.2019.e01838

Vargha, A., & Delaney, H. D. (2000). A critique and improvement of the CL common language effect size statistics of McGraw and Wong. *Journal of Educational and Behavioral Statistics, 25*(2), 101–132. doi:10.3102/10769986025002101

### Examples

```
# Example 1: Using Vargha and Delaney rules:
cle = 0.23
th_cle(cle)

# Example 2: Convert to rank-biserial and use Sawilowsky rules:
cle = 0.23
th_cle(cle, qual="sawilowsky", convert="rb")
```

---

## `th_cohen_d` *Rules of Thumb for Cohen d*

### Description

This function will give a qualification (classification) for Cohen d

### Usage

```
th_cohen_d(d, qual = "sawilowsky")
```

### Arguments

| | |
|---|---|
| `d` | the Cohen d value |
| `qual` | optional the rule of thumb to be used. Either "sawilowsky" (default), "brydges", cohen", "rosenthal", or "lovakov" |

**Details**

The following rules-of-thumb can be used:

*"brydges"* => Brydges (2019, p. 5):

| \|d\| | Interpretation |
|---|---|
| 0.00 < 0.15 | negligible |
| 0.15 < 0.40 | small |
| 0.40 < 0.75 | medium |
| 0.75 or more | large |

"cohen" => Cohen (1988, p. 40)

| \|d\| | Interpretation |
|---|---|
| 0.00 < 0.20 | negligible |
| 0.20 < 0.50 | small |
| 0.50 < 0.80 | medium |
| 0.80 or more | large |

"sawilowsky" => Sawilowsky (2009, p. 599)

| \|d\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.20 | very small |
| 0.20 < 0.50 | small |
| 0.50 < 0.80 | medium |
| 0.80 < 1.20 | large |
| 1.20 < 2.00 | very large |
| 2.00 or more | huge |

"lovakov" => Lovakov and Agadullina (2021, p. 501)

| \|d\| | Interpretation |
|---|---|
| 0.00 < 0.15 | negligible |
| 0.15 < 0.36 | small |
| 0.36 < 0.65 | medium |
| 0.65 or more | large |

"rosenthal" => Rosenthal (1996, p. 45)

| \|d\| | Interpretation |
|---|---|
| 0.00 < 0.20 | negligible |
| 0.20 < 0.50 | small |
| 0.50 < 0.80 | medium |
| 0.80 < 1.30 | large |
| 0.80 or more | very large |

### Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

### Before, After and Alternatives

Cohen d for one-sample and Hedges g could be converted to Cohen d: `es_convert`, for the conversions. `es_cohen_d`, to determine Cohen d' `es_hedges_g_os`, for Hedges g

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Brydges, C. R. (2019). Effect size guidelines, sample size calculations, and statistical power in gerontology. *Innovation in Aging, 3*(4), 1–8. doi:10.1093/geroni/igz036

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Lovakov, A., & Agadullina, E. R. (2021). Empirically derived guidelines for effect size interpretation in social psychology. *European Journal of Social Psychology, 51*(3), 485–504. doi:10.1002/ejsp.2752

Rosenthal, J. A. (1996). Qualitative descriptors of strength of association and effect size. *Journal of Social Service Research, 21*(4), 37–59. doi:10.1300/J079v21n04_02

Sawilowsky, S. (2009). New effect size rules of thumb. *Journal of Modern Applied Statistical Methods, 8*(2). doi:10.22237/jmasm/1257035100

### Examples

```
es = 0.6
th_cohen_d(es)
```

---

## `th_cohen_f` *Rule-of-Thumb for Cohen f*

### Description

Simple function to use a rule-of-thumb for the Cohen f effect size.

### Usage

```
th_cohen_f(f, qual = "cohen")
```

### Arguments

| | |
|---|---|
| `f` | the Cohen f value |
| `qual` | optional setting for which rule of thumb to use. Currently only "cohen" |

### Details

Cohen's rule of thumb for Cohen f (1988, pp. 285-287):

| \|f\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.25 | small |
| 0.25 < 0.40 | medium |
| 0.40 or more | large |

### Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

### See Also

`es_cohen_f`, to determine Cohen f

`r_rosenthal`, to determine the Rosenthal correlation, which Cohen called also Cohen f

---

## `th_cohen_g` *Rule-of-Thumb for Cohen g*

### Description

Simple function to use a rule-of-thumb for the Cohen g effect size.

This function is shown in this YouTube video and the effect size is also described at PeterStatistics.com

### Usage

```
th_cohen_g(g, qual = "cohen")
```

### Arguments

| | |
|---|---|
| `g` | the Cohen g value |
| `qual` | optional setting for which rule of thumb to use. Currently only "cohen" |

### Details

Cohen's rule of thumb for Cohen g (1988, pp. 147-149):

| \|g\| | Interpretation |
|---|---|
| 0.00 < 0.05 | negligible |
| 0.05 < 0.15 | small |
| 0.15 < 0.25 | medium |
| 0.25 or more | large |

### Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

### Before, After and Alternatives

`es_cohen_g`, to determine Cohen g

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

### Examples

```
es = 0.6
th_cohen_g(es)
```

---

## `th_cohen_h` *Rule-of-Thumb for Cohen h*

### Description

Simple function to use a rule-of-thumb for the Cohen h effect size.

This function is shown in this YouTube video and the effect size is also described at PeterStatistics.com

### Usage

```
th_cohen_h(h, qual = "cohen")
```

## Arguments

| | |
|---|---|
| `h` | the Cohen h value |
| `qual` | optional setting for which rule of thumb to use. Currently only 'cohen' |

## Details

Cohen's rule of thumb for Cohen g (1988, p. 198):

| \|h\| | Interpretation |
|---|---|
| 0.00 < 0.20 | negligible |
| 0.20 < 0.50 | small |
| 0.50 < 0.80 | medium |
| 0.80 or more | large |

Note that Cohen actually just lists small = 0.20, medium = 0.50, and large = 0.80.

## Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

## Before, After and Alternatives

`es_cohen_h`, to determine Cohen h `es_cohen_h_os`, to determine Cohen h', then use `es_convert`

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

## Examples

```
es = 0.6
th_cohen_d(es)
```

---

`th_cohen_w` *Rule-of-Thumb for Cohen w*

---

## Description

Simple function to use a rule-of-thumb for the Cohen w effect size.

The measure is also described at PeterStatistics.com

## Usage

```
th_cohen_w(w, qual = "cohen")
```

## Arguments

| | |
|---|---|
| `w` | the Cohen w value |
| `qual` | optional setting for which rule of thumb to use. Currently only 'cohen' |

## Details

Cohen's rule of thumb for Cohen w (1988, p. 227):

| \|w\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.30 | small |
| 0.30 < 0.50 | medium |
| 0.50 or more | large |

## Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

## Before, After and Alternatives

Before using this function you need to obtain a Cohen w value: `es_cohen_w`, to determine Cohen w

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

## Examples

```
es = 0.6
th_cohen_w(es)
```

---

## `th_cramer_v` *Rule-of-Thumb for Cramér V*

---

### Description

Simple function to use a rule-of-thumb for the Cramér V effect size. Note however that many will actually use the rule-of-thumb for Cohen w and convert Cramér V to Cohen w first.

The measure is also described at PeterStatistics.com

### Usage

```
th_cramer_v(v, qual = "rea-parker")
```

### Arguments

| | |
|---|---|
| `v` | the Cramér V value |
| `qual` | optional setting for which rule of thumb to use. Either "rea-parker" (default), "akoglu", "calamba-rustico" |

### Details

*"rea-parker"* => Uses Rea and Parker (1992, p. 203):

| \|v\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.20 | weak |
| 0.20 < 0.40 | moderate |
| 0.40 < 0.60 | relatively strong |
| 0.60 < 0.80 | strong |
| 0.80 or more | very strong |

*"akoglu"* => Uses Akoglu (2018, p. 92):

| \|v\| | Interpretation |
|---|---|
| 0.00 < 0.05 | very weak |
| 0.05 < 0.10 | weak |
| 0.10 < 0.15 | moderate |
| 0.15 < 0.25 | strong |
| 0.25 or more | very strong |

*"calamba-rustico"* => Uses Calamba and Rustico (2019, p. 7):

| \|v\| | Interpretation |
|---|---|
| 0.00 < 0.15 | very weak |
| 0.15 < 0.20 | weak |
| 0.20 < 0.25 | moderate |
| 0.25 < 0.30 | moderately strong |
| 0.30 < 0.35 | strong |

| | |
|---|---|
| 0.35 < 0.50 | worrisomely strong |
| 0.50 or more | redundant |

Note that the original source has a gap from 0.40 < 0.50, I added this to the 'worrisomely strong' category.

### Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

### Before, After and Alternatives

Before using this function you need to obtain a Cramer v value: `es_cramer_v_gof`, to determine Cramér V for a Goodness-of-Fit test. `es_cramer_v_ind`, to determine Cramér V for a test of independence.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Akoglu, H. (2018). User's guide to correlation coefficients. *Turkish Journal of Emergency Medicine, 18*(3), 91–93. doi:10.1016/j.tjem.2018.08.001

Calamba, S. S., & Rustico, E. M. P. (2019). Usefulness of code of ethics for professional accountants in resolving ethical conflicts in the Philippines.

Rea, L. M., & Parker, R. A. (1992). *Designing and conducting survey research: A comprehensive guide*. Jossey-Bass Publishers.

### Examples

```
es = 0.6
th_cramer_v(es)
```

---

## `th_kaiser_b` *Rule-of-Thumb for Kaiser b*

### Description

Simple function to use a rule-of-thumb for the Kaiser b variation measure.

### Usage

```
th_kaiser_b(b, qual = "kaiser")
```

### Arguments

| | |
|---|---|
| `b` | the Cohen b value |
| `qual` | optional setting for which rule of thumb to use. Currently only "kaiser" |

### Details

Kaiser's rule of thumb for Kaiser b (1968, p. 212):

| \|b\| | Interpretation |
|---|---|
| 0.00 < 0.70 | terrible |
| 0.70 < 0.80 | poor |
| 0.80 < 0.90 | fair |
| 0.90 < 0.95 | good |
| 0.95 < 1.00 | excellent |

### Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Kaiser, H. F. (1968). A measure of the population quality of legislative apportionment. *American Political Science Review, 62*(1), 208–215. doi:10.2307/1953335

### See Also

`me_qv`, to determine Kaiser b

---

## `th_odds_ratio` *Rules of thumb for Odds Ratio*

### Description

This function will give a qualification (classification) for a given Odds Ratio

### Usage

```
th_odds_ratio(or, qual = "chen")
```

### Arguments

| | |
|---|---|
| `or` | the odds ratio |
| `qual` | optional rule of thumb to use. Either "chen" (default), "wuensch", "jones1", "jones2", or "hopkins" |

### Details

If the OR is less than 1, the alternative is used, i.e. 1/OR.

"chen" => Chen et al. (2010, p. 864)

| OR | Interpretation |
|---|---|
| 1.00 < 1.68 | negligible |
| 1.68 < 3.47 | weak |
| 3.47 < 6.71 | moderate |
| 6.71 or more | strong |

"hopkins" => Hopkins (1997, tbl. 1)

| \(OR^\ast\) | Interpretation |
|---|---|
| 1.00 < 1.50 | trivial |
| 1.50 < 3.50 | small |
| 3.50 < 9.00 | moderate |
| 9.00 < 32.0 | large |
| 32.0 < 360 | very large |
| 360 or more | nearly perfect |

"jones1" => Jones (2014)

| OR | Interpretation |
|---|---|
| 1.00 < 1.5 | negligible |
| 1.5 < 2.5 | small |
| 2.5 < 4.3 | medium |
| 4.3 or more | large |

"jones2" => Jones (2014)

| OR | Interpretation |
|---|---|
| 1.00 < 1.5 | negligible |
| 1.5 < 3.5 | small |
| 2.5 < 9.0 | medium |
| 9.0 or more | large |

"wuensch" => Wuensch (2009, p. 2)

| OR | Interpretation |
|---|---|
| 1.00 < 1.49 | negligible |
| 1.49 < 3.45 | small |
| 3.45 < 9 | medium |
| 9 or more | large |

## Value

Dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Chen, H., Cohen, P., & Chen, S. (2010). How big is a big Odds Ratio? Interpreting the magnitudes of Odds Ratios in epidemiological studies. *Communications in Statistics - Simulation and Computation, 39*(4), 860–864. doi:10.1080/03610911003650383

Hopkins, W. G. (2006, August 7). New view of statistics: Effect magnitudes. http://www.sportsci.org/resource/stats/effec

Jones, K. (2014, June 5). How do you interpret the odds ratio (OR)? ResearchGate. https://www.researchgate.net/post/Ho

Wuensch, K. (2009). Cohen's conventions for small, medium, and large effects. https://imaging.mrc-cbu.cam.ac.uk/statswiki/FAQ/effectSize?action=AttachFile&do=get&target=esize.doc

## Examples

```
th_odds_ratio(5.23)
th_odds_ratio(5.23, qual="wuensch")
```

---

# `th_pearson_r` *Rules of Thumb for Pearson Correlation Coefficient*

## Description

This function will give a qualification (classification) to a given correlation coefficient

## Usage

```
th_pearson_r(r, qual = "bartz")
```

## Arguments

| | |
|---|---|
| `r` | the correlation coefficient |
| `qual` | optional the rule of thumb to be used. Either `"bartz"` (default), `"agnes"`, `"brydges"`, `"cohen"`, `"disha"`, `"funder"`, `"hopkins"`, `"lovakov"`, `"rafter"`, `"rea"`, `"rosenthal"`, `"rumsey"`, `"gignac"`, or `"hemphill"` |

### Details

The following rules-of-thumb can be used:

"agnes" => Agnes (2011)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.20 | negligible |
| 0.20 < 0.40 | low |
| 0.40 < 0.60 | moderate |
| 0.60 < 0.80 | marked |
| 0.80 or more | high |

"bartz" => Bartz (1988, p. 199)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.20 | very low |
| 0.20 < 0.40 | low |
| 0.40 < 0.60 | moderate |
| 0.60 < 0.80 | strong |
| 0.80 or more | very high |

"brydges" => Brydges (2019, p. 5) =

"gignac" => Gignac and Szodorai (2016, p. 75) =

"hemphill" => Hemphill (2003, p. 78)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.20 | small |
| 0.20 < 0.30 | medium |
| 0.30 or more | large |

"cohen" => Cohen (1988, p. 82)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.20 | negligible |
| 0.20 < 0.50 | small |
| 0.50 < 0.80 | medium |
| 0.80 or more | large |

"disha" => Disha (2016)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.10 | markedly low and negligible |
| 0.10 < 0.30 | very low |
| 0.30 < 0.50 | low |
| 0.50 < 0.70 | moderate |
| 0.70 < 0.90 | high |

| | |
|---|---|
| 0.90 or more | very high |

"funder" => Funder and Ozer (2019, p. 166)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.05 | negligible |
| 0.05 < 0.10 | very small |
| 0.10 < 0.20 | small |
| 0.20 < 0.30 | medium |
| 0.30 < 0.40 | large |
| 0.40 or more | very large |

"hopkins" => Hopkins (2006, tbl. 1)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.10 | trivial |
| 0.10 < 0.30 | low |
| 0.30 < 0.50 | moderate |
| 0.50 < 0.70 | high |
| 0.70 < 0.90 | very large |
| 0.90 or more | nearly perfect |

"lovakov" => Lovakov and Agadullina (2021, p. 514)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.12 | negligible |
| 0.12 < 0.24 | small |
| 0.24 < 0.41 | medium |
| 0.41 or more | large |

"rafter" => Rafter et al. (2003, p. 194)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.25 | weak |
| 0.25 < 0.75 | moderate |
| 0.75 or more | strong |

"rea" => Rea and Parker (2014, pp. 229, 271)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.30 | low |
| 0.30 < 0.60 | moderate |
| 0.60 < 0.75 | strong |
| 0.75 or more | very strong |

"rosenthal" => Rosenthal (1996, p. 45)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.10 | negligible |
| 0.10 < 0.30 | small |
| 0.30 < 0.50 | medium |
| 0.50 < 0.70 | large |
| 0.70 or more | very large |

"rumsey" => Rumsey (2011, p. 284)

| \|r\| | Interpretation |
|---|---|
| 0.00 < 0.30 | negligible |
| 0.30 < 0.50 | weak |
| 0.50 < 0.70 | moderate |
| 0.70 or more | strong |

## Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Agnes. (2011, April 16). Correlation – Correlation coefficient, r. Finance Training Course. https://financetrainingcourse.com/education/2011/04/correlation-correlation-coefficient-r/

Bartz, A. E. (1988). *Basic statistical concepts* (3rd ed.). Macmillan.

Brydges, C. R. (2019). Effect size guidelines, sample size calculations, and statistical power in gerontology. *Innovation in Aging, 3*(4), 1–8. doi:10.1093/geroni/igz036

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). L. Erlbaum Associates.

Disha, M. (2016, November 3). Correlation: Meaning, types and its computation. Your Article Library. https://www.yourarticlelibrary.com/statistics-2/correlation-meaning-types-and-its-computation-statistics/92001

Funder, D. C., & Ozer, D. J. (2019). Evaluating effect size in psychological research: Sense and nonsense. *Advances in Methods and Practices in Psychological Science, 2*(2), 156–168. doi:10.1177/2515245919847202

Gignac, G. E., & Szodorai, E. T. (2016). Effect size guidelines for individual differences researchers. *Personality and Individual Differences, 102*, 74–78. doi:10.1016/j.paid.2016.06.069

Hemphill, J. F. (2003). Interpreting the magnitudes of correlation coefficients. *American Psychologist, 58*(1), 78–79. doi:10.1037/0003-066X.58.1.78

Hopkins, W. G. (2006, August 7). New view of statistics: Effect magnitudes. http://www.sportsci.org/resource/stats/effec

Lovakov, A., & Agadullina, E. R. (2021). Empirically derived guidelines for effect size interpretation in social psychology. *European Journal of Social Psychology, 51*(3), 485–504. doi:10.1002/ejsp.2752

Rafter, J. A., Abell, M. L., & Braselton, J. P. (2003). *Statistics with Maple*. Academic Press.

Rea, L. M., & Parker, R. A. (2014). *Designing and conducting survey research: A comprehensive guide* (4th ed.). Jossey-Bass, a Wiley brand.

Rosenthal, J. A. (1996). Qualitative descriptors of strength of association and effect size. *Journal of Social Service Research, 21*(4), 37–59. doi:10.1300/J079v21n04_02

Rumsey, D. J. (2011). *Statistics for dummies* (2nd ed.). Wiley.

## See Also

`r_pearson`, to determine Pearson correlation coefficient or

`r_rosenthal`, to determine Rosenthal correlation coefficient

## Examples

```
es = 0.6
th_pearson_r(es)
```

---

# `th_post_hoc_gof` *Post-Hoc Goodness-of-Fit Rules-of-Thumb*

## Description

This function will add a classification to the results of **es_post_hoc_gof()** using a rules-of-thumb. This is frowned upon by some, and the rule-of-thumb can vary per discipline.

## Usage

```
th_post_hoc_gof(eff_sizes, convert = FALSE, ph_results = NULL, ...)
```

## Arguments

| | |
|---|---|
| `eff_sizes` | dataframe, the dataframe from es_post_hoc_gof() |
| `convert` | boolean, optional. convert the effect size to use the rule-of-thumb from another, see details |
| `ph_results` | dataframe, optional. the post-hoc analysis results, required for JBM-E and Fei. |
| `...` | optional. additional arguments for the specific rule-of-thumb that are passed along. Most common 'qual=...' for a specific set of rules-of-thumb. |

## Details

For Johnston-Berry-Mielke E and Fei, a conversion is always done to Cramér V, when setting *convert=True* it will convert it again to Cohen w.

Other possible conversions are Cohen h' to Cohen h, and Cramér V to Cohen w.

See the separate documentation for each of the rules-of-thumb, or conversion.

## Value

df, dataframe with the same dataframe as the provided *eff_sizes*, but added:

| | |
|---|---|
| `qualification` | , the qualification using the rule-of-thumb. |
| `reference` | , a reference to the source for the rule-of-thumb. |

If a conversion was done or needed:

| | |
|---|---|
| `conversion description` | , the value of the converted measure |

## Before, After and Alternatives

Before using this the post-hoc effect sizes need to be made: `es_post_hoc_gof`, to obtain post-hoc effect sizes.

Depending on the measure, the function will use the rules of thumb for: `th_cohen_g`, for Cohen g. `th_cohen_h`, for Cohen h. `th_cohen_w`, for Cohen w. `th_cramer_v`, for Cramer V. `th_pearson_r`, for Pearson r.

It can also convert using: `es_convert`, to convert various effect sizes.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

# `th_rank_biserial` *Rule-of-Thumb for Rank Biserial Correlation*

## Description

Simple function to use a rule-of-thumb for the Rank Biserial Correlation.

This function is shown in this YouTube video and the measure is also described at PeterStatistics.com

## Usage

```
th_rank_biserial(rb, qual = "cohen")
```

## Arguments

| | |
|---|---|
| `rb` | the rank-biserial correlation value |
| `qual` | optional setting for which rule of thumb to use. Either "cohen" (default), "vd", "sawilowsky", "cohen-conv", "lovakov", "rosenthal", "brydges" |

## Details

Cohen's rule of thumb for rank-biserial correlation (1988, p. 82):

| \|r_b\| | Interpretation |
|---|---|
| 0.000 < 0.125 | negligible |
| 0.125 < 0.304 | small |
| 0.304 < 0.465 | medium |
| 0.465 or more | large |

Vargha and Delaney (2000, p. 106):

| \|r_b\| | Interpretation |
|---|---|
| 0.00 < 0.11 | negligible |
| 0.11 < 0.28 | small |
| 0.28 < 0.43 | medium |
| 0.43 or more | large |

## Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

## Before, After and Alternatives

Before this you might want to obtain the measure: `r_rank_biserial_is`, to determine a the rank biserial for independent samples. `r_rank_biserial_os`, to determine a the rank biserial for one-sample.

The function uses the convert function and corresponding rules of thumb: `es_convert`, to convert this to Cohen d. `th_cohen_d`, rules of thumb for Cohen d.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). L. Erlbaum Associates.

## Examples

```
# Example 1: using Cohen's rules:
rb = 0.6
th_rank_biserial(rb)

# Example 2: Convert to Cohen d, then use Cohen d rules:
rb= 0.23
th_rank_biserial(rb, qual="cohen-conv")
```

---

`th_yule_q` *Rules of thumb for Yule Q*

---

## Description

Simple function to use a rule-of-thumb for Yule Q effect size.

## Usage

```
th_yule_q(q, qual = "glen")
```

## Arguments

| | |
|---|---|
| `q` | the Yule Q value |
| `qual` | optional for which rule-of-thumb to use. Currently only "glen" |

## Details

Glen rule of thumb for Yule Q (2017):

| \|Q\| | Interpretation |
|---|---|
| 0.00 < 0.30 | negligible |
| 0.30 < 0.50 | moderate |
| 0.50 < 0.70 | substantial |
| 0.70 or more | very strong |

## Value

A dataframe with:

| | |
|---|---|
| `classification` | the qualification of the effect size |
| `reference` | a reference for the rule of thumb used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Glen, S. (2017, August 16). Gamma Coefficient (Goodman and Kruskal's Gamma) & Yule's Q. Statistics How To. https://www.statisticshowto.com/gamma-coefficient-goodman-kruskal/

## See Also

`es_bin_bin`, to determine Yule Q

## Examples

```
q = 0.6
th_yule_q(q)
```

---

`ts_alexander_govern_owa`

*Alexander-Govern Test*

---

## Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

Schneider and Penfield (1997) looked at the Welch, Alexander-Govern and the James test (they ignored the Brown-Forsythe since they found it to perform worse than Welch or James), and concluded: “Under variance heterogeneity, Alexander-Govern’s approximation was not only comparable to the Welch test and the James second-order test but was superior, in certain instances, when coupled with the power results for those tests” (p. 285).

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

## Usage

```
ts_alexander_govern_owa(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used (Alexander & Govern, 1994, pp. 92-94):

$$A = \sum_{j=1}^{k} z_j^2$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(A, df\right)$$

With:

$$z_j = c_j + \frac{c_j^3 + 3\times c_j}{b_j} - \frac{4\times c_j^7 + 33\times c_j^5 + 240\times c_j^3 + 855\times c_j}{10\times b_j^2 + 8\times b_j\times c_j^4 + 1000\times b_j}$$

$$c_j = \sqrt{a_j\times\ln\left(1 + \frac{t_j^2}{n_j - 1}\right)}$$

$$b_j = 48\times a_j^2$$

$$a_j = n_j - 1.5$$

$$t_j = \frac{\bar{x}_j - \bar{y}_w}{\sqrt{\frac{s_j^2}{n_j}}}$$

$$\bar{y}_w = \sum_{j=1}^{k} h_j \times \bar{x}_j$$

$$h_j = \frac{w_j}{w}$$

$$w_j = \frac{n_j}{s_j^2}$$

$$w = \sum_{j=1}^{k} w_j$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j} \left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{j=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols:*

- $n$ the total sample size
- $k$ the number of categories
- $x_{i,j}$ the i-th score in category j
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $df$ the degrees of freedom
- $\chi^2\left(\dots,\dots\right)$ the cumulative distribution function of the chi-square distribution.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Alexander, R. A., & Govern, D. M. (1994). A new and simpler approximation for ANOVA under variance heterogeneity. *Journal of Educational Statistics, 19*(2), 91–101. doi:10.2307/1165140

Schneider, P. J., & Penfield, D. A. (1997). Alexander and Govern's approximation: Providing an alternative to ANOVA under variance heterogeneity. *The Journal of Experimental Education, 65*(3), 271–286. doi:10.1080/00220973.1997.9943459

---

## ts_bhapkar *Bhapkar Test*

---

### Description

If you are only interested if the overall distribution changed (i.e. if the percentages from each category changed or not), you can perform a marginal homogeneity test. There are two that seem to be quite popular for this, the Stuart-Maxwell test (Stuart, 1955; Maxwell, 1970), and the Bhapkar test (Bhapkar, 1961; 1966). According Uebersax (2006) (which also has a nice example) the Bhapkar one is preferred.

Simply put, a marginal homogeneity test, looks at the row vs column proportions. Since in a paired test, the options are the same, if the row and column proportions are the same, nothing changed between the two variables.

### Usage

```
ts_bhapkar(field1, field2, categories = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |

### Details

The formula used is:

$$\chi_B^2 = n \times d' \times S^{-1} \times d$$

With:

$$S_{i,i} = p_{i,.} + p_{.,i} - 2 \times p_{i,i} - \left(p_{i,.} - p_{.,i}\right)^2$$

$$S_{i,j} = -\left(p_{i,j} + p_{j,i}\right) - \left(p_{i,.} - p_{.,i}\right) \times \left(p_{j,.} - p_{.,j}\right)$$

$$d_i = p_{i,.} - p_{.,i}$$

$$p_{i,j} = \frac{F_{i,j}}{n}$$

$$d = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_{r-1} \end{bmatrix}$$

$$S = \begin{bmatrix} S_{1,1} & S_{1,2} & \dots & S_{1,c-1} \\ S_{2,1} & S_{2,2} & \dots & S_{2,c-1} \\ \dots & \dots & \dots & \dots \\ S_{r-1,1} & S_{r-1,2} & \dots & S_{r-1,c-1} \end{bmatrix}$$

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$

The p-value (sig.):

$$df = r - 1 = c - 1$$

$$sig. = 1 - \chi^2\left(\chi_B^2, df\right)$$

*Symbols used:*

- $r$, is the number of rows (categories in the first variable)
- $c$, is the number of columns (categories in the second variable)
- $n$, is the total number of scores
- $F_{i,j}$, is the frequency (count) of scores equal to the i-th category in the first variable, and the j-th category in the second.
- $p_{i,.}$, The sum of the proportions in row i
- $p_{.,i}$, The sum of the proportions in column i
- $d'$, is the transpose of the d vector
- $S^{-1}$, is the inverse of the S matrix.
- $\chi^2\left(\dots\right)$, the cumulative distribution function for the chi-square distribution.

*Note*

- The d vector and S matrix are one row (and column) less.
- This test only differs from the Stuart-Maxwell test in the calculation of S
- The test was introduced by Bhapkar (1961, 1966)

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the chi-squared value |
| `df` | the degrees of freedom used in the test |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Bhapkar, V. P. (1961). Some tests for categorical data. *The Annals of Mathematical Statistics, 32*(1), 72–83. doi:10.1214/aoms/1177705140

Bhapkar, V. P. (1966). A note on the equivalence of two test criteria for hypotheses in categorical data. *Journal of the American Statistical Association, 61*(313), 228–235. doi:10.1080/01621459.1966.10502021

Maxwell, A. E. (1970). Comparing the classification of subjects by two independent judges. *The British Journal of Psychiatry, 116*(535), 651–655. doi:10.1192/bjp.116.535.651

Stuart, A. (1955). A test for homogeneity of the marginal distributions in a two-way classification. *Biometrika, 42*(3/4), 412–416. doi:10.2307/2333387

Uebersax, J. (2006, August 30). McNemar tests of marginal homogeneity. http://www.john-uebersax.com/stat/mcnemar.htm

---

## `ts_binomial_os` *One-Sample Binomial Test*

---

### Description

Performs a one-sample (exact) binomial test.

This test can be useful with a single binary variable as input. The null hypothesis is usually that the proportions of the two categories in the population are equal (i.e. 0.5 for each). If the p-value of the test is below the pre-defined alpha level (usually 5% = 0.05) the null hypothesis is rejected and the two categories differ in proportion significantly.

The input for the function doesn't have to be a binary variable. A nominal variable can also be used and the two categories to compare indicated.

A significance in general is the probability of a result as in the sample, or more extreme, if the null hypothesis is true. For a two-tailed binomial test the 'or more extreme' causes a bit of a complication. There are different methods to approach this problem. See the details for more information.

A YouTube video on the binomial test.

This function is shown in this YouTube video and the binomial test is also described at PeterStatistics.com

### Usage

```
ts_binomial_os(
  data,
  p0 = 0.5,
  p0Cat = NULL,
  codes = NULL,
  twoSidedMethod = c("eqdist", "double", "smallp")
)
```

### Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `p0` | Optional hypothesized proportion for the first category (default is 0.5) |
| `p0Cat` | Optional the category for which p0 was used |
| `codes` | Optional vector with the two codes to use |
| `twoSidedMethod` | Optional method to be used for 2-sided significance (see details) |

### Details

To decide on which category is associated with p0 the following is used:

- If codes are provided, the first code is assumed to be the category for the p0.
- If p0Cat is specified that will be used for p0 and all other categories will be considered as category 2, this means if there are more than two categories the remaining two or more (besides p0Cat) will be merged as one large category.
- If neither codes or p0Cat is specified and more than two categories are in the data a warning is printed and no results.

- If neither codes or p0Cat is specified and there are two categories, p0 is assumed to be for the category closest matching the p0 value (i.e. if p0 is above 0.5 the category with the highest count is assumed to be used for p0)

For the formulas below it is assumed that the observed proportion is less than the expected proportion, if this isn't the case, the right-tail probabilities are used. A one sided p-value is calculated first:

$$sig_{one-tail} = \text{Bin}\left(n, n_{min}, p_0^*\right)$$

With:

$$n_min = min\left\{n_s, n_f\right\}$$

$$p_0^* = \begin{cases} p_0 & \text{if } n_{min} = n_s \\ 1 - p_0 & \text{if } n_{min} = n_f \end{cases}$$

*Symbols used:*

- $n$ is the number of cases
- $n_s$ is the number of successes
- $n_f$ is the number of failures
- $p_0$ is the probability of a success according to the null hypothesis
- $p_0^*$ is the probability adjusted in case failures is used
- $\text{Bin}\left(\dots\right)$ the binomial cumulative distribution function

For the two sided significance three options can be used.

*Option 1: Equal Distance Method (twoSidedMethod="eqdist")*

$$sig_{two-tail} = B\left(n, n_{min}, p_0^*\right) + 1 - B\left(n, \lfloor 2 \times n_0 \rfloor - n_{min} - 1, p_0^*\right)$$

With:

$$n_0 = \lfloor n \times p_0 \rfloor$$

This method looks at the number of cases. In a sample of $n$ people, we'd then expect $n_0 = \lfloor n \times p_0 \rfloor$ successes (we round the result down to the nearest integer). We only had $n_{min}$, so a difference of $n_0 - n_{min}$. The 'equal distance method' now means to look for the chance of having $k$ or less, and $n_0 + n_0 - n_{min} = 2 \times n_0 - n_min$ or more. Each of these two probabilities can be found using a binomial distribution. Adding these two together than gives the two-sided significance.

*Option 2: Small p-method (twoSidedMethod="smallp")*

$$sig_{two-tail} = B\left(n, n_{min}, p_0^*\right) + \sum_{i=n_{min}+1}^{n} \begin{cases} 0 & \text{if } b\left(n, i, p_0^*\right) > b\left(n, n_{min}, p_0^*\right) \\ b\left(n, i, p_0^*\right) & \text{if } x \leq b\left(n, i, p_0^*\right) > b\left(n, n_{min}, p_0^*\right) \end{cases}$$

With: $b\left(\dots\right)$ as the binomial probability mass function.

This method looks at the probabilities itself. $b\left(n, n_{min}, p_0^*\right)$ is the probability of having exactly $n_{min}$ out of a group of n, with a chance $p_0^*$ each time. The method of small p-values now considers 'or more extreme' any number between 0 and n (the sample size) that has a probability less or equal to this. This means we need to go over each option, determine the probability and check if it is lower or equal. So, the probability of 0 successes, the probability of 1 success, etc. The sum for all of those will be the two-sided significance. We can reduce the work a little since any value below $n_{min}$ , will also have a lower probability, so we only need to sum over the ones above it and add the one-sided significance to the sum of those.

*Option 3: Double single (twoSidedMethod="double")*

$$sig_{two-tail} = 2 \times sig_{one-tail}$$

Fairly straight forward. Just double the one-sided significance.

### Value

Dataframe with:

| | |
|---|---|
| `pValue` | two-sided p-value |
| `test` | a description of the test used |

### Before, After and Alternatives

Before running the test you might first want to get an impression using a frequency table: tab_frequency

After the test you might want an effect size measure: `es_cohen_g`, for Cohen g `es_cohen_h_os`, for Cohen h' `es_alt_ratio`, for Alternative Ratio

Alternatives for this test could be: `ts_score_os`, for One-Sample Score Test `ts_wald_os`, for One-Sample Wald Test

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### Examples

```
#Example 1: Numeric list
ex1 = c(1, 1, 2, 1, 2, 1, 2, 1)
ts_binomial_os(ex1)
ts_binomial_os(ex1, p0=0.3)

#Example 2: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ts_binomial_os(df1['sex'])
ts_binomial_os(df1['mar1'], codes=c("DIVORCED", "NEVER MARRIED"))
```

---

## `ts_box_owa` *Box F-Test*

### Description

Tests if the means (averages) of each category could be the same in the population.

Box proposed a correction to the original Fisher one-way ANOVA, on both the test-statistic and the degrees of freedom.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

### Usage

```
ts_box_owa(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

### Details

The formula used (Box, 1954, p. 299):

$$F_B = \frac{F_F}{c}$$

$$df_1 = \frac{\left(\sum_{j=1}^{k}\left(n - n_j\right) \times s_j^2\right)^2}{\left(\sum_{j=1} n_j \times s_j^2\right)^2 + n \times \sum_{j=1}^{k}\left(n - 2 \times n_j\right) \times s_j^4}$$

$$df_2 = \frac{\left(\sum_{j=1}^{k}\left(n_j - 1\right) \times s_j^2\right)^2}{\sum_{j=1}^{k}\left(n_j - 1\right) \times s_j^4}$$

$$sig. = 1 - F\left(F_B, df_1, df_2\right)$$

With:

$$c = \frac{n - k}{n \times (k - 1)} \times \frac{\sum_{j=1}^{k}\left(n - n_j\right) \times s_j^2}{\sum_{j=1}^{k}\left(n_j - 1\right) \times s_j^2}$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

$$n = \sum_{j=1}^{k} n_j$$

*Symbols:*

- $F - F$ the F statistic of the classic/Fisher one-way ANOVA. See *ts_fisher_owa()* for details.
- $n$ the total sample size
- $k$ the number of categories
- $x_{i,j}$ the i-th score in category j
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $df$ the degrees of freedom
- $F\left(\ldots, \ldots, \ldots\right)$ the cumulative distribution function of the F distribution.

This also appears to give the same results for the test statistic, df_1 as the Brown-Forsythe test for means but a different df_2

The *doex* and *onewaytests* libraries used to have a different method for calculating df_1 but after personal communication with the creators of those packages they mentioned to fix it in an update.

Asiribo and Gurland (1990) derive the same correction as Box, although their notation for $df_1^*$ is different, but will give the same result.

### Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the test statistic (F value) |
| `df1` | the degrees of freedom 1 |
| `df2` | the degrees of freedom 2 |
| `pValue` | the significance (p-value) |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Asiribo, O., & Gurland, J. (1990). Coping with variance heterogeneity. *Communications in Statistics - Theory and Methods, 19*(11), 4029–4048. doi:10.1080/03610929008830427

Box, G. E. P. (1954). Some theorems on quadratic forms applied in the study of analysis of variance problems, I: Effect of inequality of variance in the one-way classification. *The Annals of Mathematical Statistics, 25*(2), 290–302. doi:10.1214/aoms/1177728786

---

## `ts_brown_forsythe_owa` *Brown-Forsythe Means Test*

### Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

### Usage

```
ts_brown_forsythe_owa(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used (Brown & Forsythe, 1974, p. 130):

$$F_{BF} = \frac{\sum_{j=1}^{k} n_j \times \left(\bar{x}_j - \bar{x}\right)^2}{\sum_{j=1}^{k} \left(1 - \frac{n_j}{n}\right) \times s_j^2}$$

$$df_1 = k - 1$$

$$df_2 = \frac{\left(\sum_{j=1}^{k} \left(1 - \frac{n_j}{n}\right) \times s_j^2\right)^2}{\sum_{j=1}^{k} \frac{\left(1 - \frac{n_j}{n}\right) \times s_j^4}{n_j - 1}}$$

$$sig. = 1 - F\left(F_{BF}, df_1, df_2\right)$$

With:

$$s_j^2 = \frac{\sum_{i=1}^{n_j} \left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

$$\bar{x} = \frac{\sum_{i=1}^{k} n_j \times \bar{x}_j}{n}$$

$$n = \sum_{j=1}^{k} n_j$$

*Symbols:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n$ the total sample size
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $df$ the degrees of freedom
- $F\left(\dots, \dots, \dots\right)$ the cumulative distribution function of the F distribution.

This appears to give the same results as the Box correction, except for $df_2$.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the test statistic (F value) |
| `df1` | the degrees of freedom 1 |
| `df2` | the degrees of freedom 2 |
| `p-value` | the significance (p-value) |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Brown, M. B., & Forsythe, A. B. (1974). The small sample behavior of some statistics which test the equality of several means. *Technometrics, 16*(1), 129–132. https://doi.org/10.1080/00401706.1974.10489158

---

## `ts_cochran_owa` *Cochran One-Way ANOVA*

### Description

Tests if the means (averages) of each category could be the same in the population.

Note that according to Hartung et al. (2002, p. 225) the Cochran test is the standard test in meta-analysis, but should not be used, since it is always too liberal.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

### Usage

```
ts_cochran_owa(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

### Details

The formula used is (Cavus & Yazıcı, 2020, p. 5; Hartung et al., 2002, p. 202; Mezui-Mbeng, 2015, p. 787):

$$\chi_C^2 = \sum_{j=1}^{k} w_j \times \left(\bar{x}_j - \bar{y}_w\right)^2$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_C^2, df\right)$$

With:

$$\bar{y}_w = \frac{\sum_{j=1}^{k} w_j \times \bar{x}_j}{\sum_{j=1}^{k} w_j} = \sum_{j=1}^{k} h_j \times \bar{x}_j$$

$$h_j = \frac{w_j}{w}$$

$$w = \sum_{j=1}^{k} w_j$$

$$w_j = \frac{n_j}{s_j^2}$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j} \left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $x_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $w_j$ the weight for category j
- $h_j$ the adjusted weight for category j
- $df$ the degrees of freedom
- $\chi^2\left(\ldots,\ldots\right)$ the cumulative distribution function of the chi-square distribution.

Couldn't really find the formula in the original article which is from Cochran (1937)

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the chi-square-statistic from the test |
| `df` | the degrees of freedom |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cavus, M., & Yazıcı, B. (2020). Testing the equality of normal distributed and independent groups' means under unequal variances by doex package. *The R Journal, 12*(2), 134. https://doi.org/10.32614/RJ-2021-008

Cochran, W. G. (1937). Problems arising in the analysis of a series of similar experiments. *Supplement to the Journal of the Royal Statistical Society, 4*(1), 102–118. https://doi.org/10.2307/2984123

Hartung, J., Argaç, D., & Makambi, K. H. (2002). Small sample properties of tests on homogeneity in one-way anova and meta-analysis. *Statistical Papers, 43*(2), 197–235. https://doi.org/10.1007/s00362-002-0097-8

Mezui-Mbeng, P. (2015). A note on Cochran test for homogeneity in two ways ANOVA and meta-analysis. *Open Journal of Statistics, 5*(7), 787–796. https://doi.org/10.4236/ojs.2015.57078

---

## `ts_cochran_q` *Cochran Q Test*

---

### Description

A test for multiple binairy variables. The null hypothesis is that the proportion of successes is the same in all groups.

If the p-value (sig.) is below a certain threshold (usually .05) the assumption is rejected and at least one category has a significant different number of successes than at least one other group, in the population.

If the test is significant (below the threshold) a post-hoc Dunn test could be used, or pairwise McNemar-Bowker.

### Usage

```
ts_cochran_q(data, success = NULL)
```

### Arguments

| | |
|---|---|
| `data` | dataframe with the binary scores |
| `success` | indicator for what is considered a success (default is first value found) |

### Details

The formula used (Cochran, 1950, p. 259):

$$Q = \frac{\left(k - 1\right) \times \sum_{j=1}^{k} \left(C_j - \bar{C}\right)^2}{k \times \sum_{i=1}^{n} R_i - \sum_{i=1}^{n} R_i^2}$$

$$sig. = 1 - \chi^2\left(Q, df\right)$$

With:

$$df = k - 1$$

*Symbols used:*

- $C_j$ the number of successes in category j
- $k$ the number of categories (factors)
- $R_i$ the number of succeses in case i
- $n$ the number of cases

**Alternatives**

*library(nonpar)*

matr = cbind(var1, var2, var3, var4)

cochrans.q(matr)

*library(RVAideMemoire)*

myData.long<-reshape(dFr, varying=c("var1", "var2", "var3", "var4"), v.names="score", timevar="var", times=c("var1", "var2", "var3", "var4"),new.row.names = 1:1000, direction="long")

cochran.qtest(score~var lid, data=myData.long)

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cochran, W. G. (1950). The comparison of percentages in matched samples. *Biometrika, 37*(3/4), 256–266. https://doi.org/10.2307/2332378

---

# `ts_cressie_read_gof` *Cressie-Read Test of Goodness-of-Fit*

## Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test, but also an exact multinomial, G-test, Freeman-Tukey, Neyman, Mod-Log Likelihood, and Freeman-Tukey-Read test are possible.

## Usage

```
ts_cressie_read_gof(
  data,
  expCount = NULL,
  cc = c("none", "yates", "pearson", "williams"),
  lambda = 2/3
)
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCount` | Optional dataframe with the categories and expected counts |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "yates2", "pearson", or "williams" |
| `lambda` | optional power to use in equation (see details) |

## Details

The formula used is (Cressie & Read, 1984, p. 442):

$$\chi_{C}^{2}=\begin{cases}2\times\sum_{i=1}^{k}\left(F_{i}\times ln\left(\frac{F_{i}}{E_{i}}\right)\right) & \text{if } \lambda=0\\ 2\times\sum_{i=1}^{k}\sum_{j=1}^{c}\left(E_{i}\times ln\left(\frac{E_{i}}{F_{i}}\right)\right) & \text{if } \lambda=-1\\ \frac{2}{\lambda\times(\lambda+1)}\times\sum_{i=1}^{k}F_{i}\times\left(\left(\frac{F_{i}}{E_{i}}\right)^{\lambda}-1\right) & \text{else}\end{cases}$$

$$df=k-1$$

$$sig.=1-\chi^{2}\left(\chi_{C}^{2},df\right)$$

With:

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

If no expected counts provided:

$$E_{i}=\frac{n}{k}$$

else:

$$E_{i}=n\times\frac{E_{p_{i}}}{n_{p}}$$

$$n_{p}=\sum_{i=1}^{k}E_{p_{i}}$$

*Symbols used:*

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies
- $n_p$ the sum of all provided expected counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

Cressie and Read (1984, p. 463) suggest to use $\lambda=\frac{2}{3}$, which is therefor the default in this function.

Note that

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

$$\chi_{CY}^{2}=\sum_{i=1}^{k}\frac{\left(\left|F_{i}-E_{i}\right|-0.5\right)^{2}}{E_{i}}$$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{CP}^{2}=\chi_{P}^{2}\times\frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{CW}^{2}=\frac{\chi_{P}^{2}}{q}$$

With:

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

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `propBelow5` | the proportion of expected counts below 5 |
| `testUsed` | a description of the test used |

## Alternatives

The *MSCquartets* library has a *powerDivStat()* function, which can return the test statistics, based on given observed and expected counts. obs = as.vector(unname(table(nomData)))

k = length(obs)

n = sum(obs)

exp = rep(1/k, k)

n*powerDivStat(obs/n, exp, lambda=2/3)

## Author(s)

P. Stikker. Companion Website, YouTube Channel

## References

Cressie, N., & Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society: Series B (Methodological), 46*(3), 440–464. doi:10.1111/j.2517-6161.1984.tb01318.x

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

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. doi:10.2307/2332518

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## See Also

Alternative tests with a nominal variable:

- `ts_pearson_gof` Pearson chi-square test of goodness-of-fit
- `ts_multinomial_gof` exact multinomial test of goodness-of-fit
- `ts_g_gof` G / Likelihood Ratio / Wilks test of goodness-of-fit
- `ts_freeman_tukey_gof` Freeman-Tukey test of goodness-of-fit
- `ts_neyman_gof` Neyman test of goodness-of-fit

- `ts_mod_log_likelihood_gof` Mod-Log Likelihood test of goodness-of-fit
- `ts_freeman_tukey_read` Freeman-Tukey-Read test of goodness-of-fit

Effect sizes that might be of interest:

- `es_cramer_v_gof` Cramér's V for goodness-of-fit
- `es_cohen_w` Cohen w
- `es_jbm_e` Johnston-Berry-Mielke E

---

## `ts_cressie_read_ind` *Cressie-Read Test of Independence / Power Divergence Test*

### Description

Cressie-Read Test of Independence / Power Divergence Test

### Usage

```
ts_cressie_read_ind(nom1, nom2, cc = NULL, lambda = 2/3)
```

### Arguments

| | |
|---|---|
| `nom1` | A vector with the data of the first variable |
| `nom2` | A vector with the data of the second variable |
| `cc` | c(NULL, "yates", "pearson", or "williams") Optional continuity correction (default is NULL) |
| `lambda` | Optional value for lambda (default is 2/3) |

### Details

The formula used is (Cressie & Read, 1984, p. 442):

$$\chi_C^2 = \begin{cases} 2 \times \sum_{i=1}^{r} \sum_{j=1}^{c} \left(F_{i,j} \times ln\left(\frac{F_{i,j}}{E_{i,j}}\right)\right) & \text{if } \lambda = 0 \\ 2 \times \sum_{i=1}^{r} \sum_{j=1}^{c} \left(E_{i,j} \times ln\left(\frac{E_{i,j}}{F_{i,j}}\right)\right) & \text{if } \lambda = -1 \\ \frac{2}{\lambda \times (\lambda+1)} \times \sum_{i=1}^{r} \sum_{j=1}^{c} F_{i,j} \times \left(\left(\frac{F_{i,j}}{E_{i,j}}\right)^{\lambda} - 1\right) & \text{else} \end{cases}$$

$$df = (r-1) \times (c-1)$$

$$sig. = 1 - \chi^2\left(\chi_C^2, df\right)$$

With:

$$n = \sum_{i=1}^{r} \sum_{j=1}^{c} F_{i,j}$$

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

*Symbols used:*

- $r$ the number of categories in the first variable (the number of rows)
- $c$ the number of categories in the second variable (the number of columns)
- $F_{i,j}$ the observed count in row i and column j
- $E_{i,j}$ the expected count in row i and column j
- $R_i$ the i-th row total
- $C_j$ the j-th column total
- $n$ the sum of all counts
- $\chi^2\left(\ldots\right)$ the chi-square cumulative density function

Cressie and Read (1984, p. 463) suggest to use $\lambda = \frac{2}{3}$, which is therefor the default in this function.

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

Use instead of $F_{i,j}$ the adjusted version defined by:

$$F_{i,j}^* = \begin{cases} F_{i,j} - 0.5 & \text{if } F_{i,j} > E_{i,j} \\ F_{i,j} & \text{if } F_{i,j} = E_{i,j} \\ F_{i,j} + 0.5 & \text{if } F_{i,j} < E_{i,j} \end{cases}$$

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_{P}^2 \times \frac{n - 1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{PW}^2 = \frac{\chi_{P}^2}{q}$$

With:

$$q = 1 + \frac{\left(n\times\left(\sum_{i=1}^r \frac{1}{R_i}\right) - 1\right) \times \left(n\times\left(\sum_{j=1}^c \frac{1}{C_j}\right) - 1\right)}{6\times n\times df}$$

## Value

dataframe with test statistic, degrees of freedom, p-value, minimum expected count, proportion of expected counts below 5, and test used

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Cressie, N., & Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society: Series B (Methodological), 46*(3), 440–464. https://doi.org/10.1111/j.2517-6161.1984.tb01318.x

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

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. https://doi.org/10.2307/2983604

## Examples

```
nom1 <- c("female", "female","female","female","female","female","female","female",
"female","female","female", "male", "male", "male", "male", "male", "male", "male",
"male", "male", "male", "male", "male", "male", "male", "male", "male","male",
"male", "male", "male", "male", "male", "male", "male", "male", "male", "male",
"male", "male", "male", "male")
nom2 <- c("nl", "nl","nl","nl","nl","nl","nl","nl", "other",
"other", "other","nl","nl","nl","nl","nl","nl","nl","nl","nl",
"nl","nl","nl","nl","nl","nl","nl","other", "other", "other",
"other", "other", "other", "other", "other", "other", "other",
"other", "other", "other", "other", "other")
ts_cressie_read_ind(nom1, nom2)
ts_cressie_read_ind(nom1, nom2, cc="yates")
ts_cressie_read_ind(nom1, nom2, cc="pearson")
ts_cressie_read_ind(nom1, nom2, cc="williams")
```

---

# `ts_fisher` *Fisher Exact test*

## Description

Perhaps the most commonly used test when you have two binary variables is the Fisher (Exact) Test (Fisher, 1922, 1950). It tests if "the relative proportions of one variable are independent of the second variable; in other words, the proportions at one variable are the same for different values of the second variable" (McDonald, 2014, p. 77).

Note that for a 2x2 table there are quite a lot of different tests. Upton (1982) discusses 24 of them. For larger tables a Fisher-Freeman-Halton Exact Test could be used.

## Usage

```
ts_fisher(field1, field2, categories1 = NULL, categories2 = NULL)
```

## Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with order for categories of field1 |
| `categories2` | : optional list with order for categories of field2 |

## Details

The formula used is from Fisher (1950, p. 96):

$$p = \sum_{i=a_{min}}^{a_{max}} \begin{cases} p_i & \text{if } p_i \leq p_s \\ 0 & \text{else} \end{cases}$$

With:

$$p_x = \frac{\binom{R_1}{x} \times \binom{n-R_1}{C_1-x}}{\binom{n}{C_1}}$$

$$a_{min} = \max\left(0, C_1 + R_1 - n\right)$$

$$a_{max} = \min\left(R_1, C_1\right)$$

$$\binom{x}{y} = \frac{x!}{y! \times (x-y)!}$$

*Symbols used:*

- $p_s$, the probability of sample cross table, i.e. p_x with x being the upper-left cell of the the cross table from the sample data.
- $R_1$, is the total of the first row,
- $C_1$ the total of the first column.
- $n$, is the total sample size.

The reason for the minimum value of 'a', is first that it cannot be negative, since these are counts. So 0 would be the lowest ever possible. However, once 'a' is set, and the totals are fixed, all other values should also be positive (or zero). The value for 'b' will be if 'a' is 0, it will simply be R1 - a. The value for 'c' is also no issue, this is simply C1 - a. However 'd' might be negative, even if a = 0. The value for 'd' is n - R1 - c. Since c = C1 - a, we get d = n - R1 - C1 + a. But this could be negative if R1 + C1 > n. So, 'a' must be at least C1 + R1 - n.

The maximum for 'a' is simply the minimum of either it's row total, or column total.

Note that $p_x$ is the probability mass function of a hypergeometric distribution.

## Value

pval : the two-sided p-value (sig.)

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Fisher, R. A. (1922). On the Interpretation of $\chi^2$ from Contingency Tables, and the Calculation of P. *Journal of the Royal Statistical Society, 85*(1), 87–94. https://doi.org/10.2307/2340521

Fisher, R. A. (1950). *Statistical methods for research workers* (11th rev.). Oliver and Boyd.

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

Upton, G. J. G. (1982). A comparison of alternative tests for the 2 x 2 comparative trial. *Journal of the Royal Statistical Society. Series A (General), 145*(1), 86–105. https://doi.org/10.2307/2981423

## See Also

`tab_cross`, to create a cross-table.

## Examples

```
#Example: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ts_fisher(df1[['mar1']], df1[['sex']], categories1=c("WIDOWED", "DIVORCED"))
```

## `ts_fisher_freeman_halton` *Fisher-Freeman-Halton Exact test*

### Description

Fisher-Freeman-Halton Exact test

### Usage

```
ts_fisher_freeman_halton(var1, var2)
```

### Arguments

| | |
|---|---|
| `var1` | A vector with the data from the first variable |
| `var2` | A vector with the data from the second variable |

### Details

This simply uses R's *fisher.test()* function from the stats library.

### Value

the two-tailed p-value/sig.

### Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

### References

Fisher, R. A. (1922). On the Interpretation of $\chi^2$ from Contingency Tables, and the Calculation of P. *Journal of the Royal Statistical Society, 85*(1), 87–94. https://doi.org/10.2307/2340521

Freeman, G. H., & Halton, J. H. (1951). Note on an exact treatment of contingency, goodness of fit and other problems of significance. *Biometrika, 38*(1/2), 141–149. https://doi.org/10.2307/2332323

### Examples

```
var1 <- c("female", "female","female","female","female","female","female","female",
          "female","female","female", "male", "male", "male", "male", "male", "male",
         "male", "male", "male", "male", "male", "male", "male", "male", "male", "male",
         "male", "male", "male", "male", "male", "male", "male", "male", "male", "male",
          "male", "male", "male", "male", "male")
var2 <- c("nl", "nl","nl","nl","nl","nl","nl","nl", "other", "other", "other",
        "nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl",
         "other", "other", "other", "other", "other", "other", "other", "other", "other",
          "other", "other", "other", "other", "other", "other")
ts_fisher(var1, var2)
```

---

## `ts_fisher_owa` *Fisher/Classic One-Way ANOVA / F-Test*

---

### Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

There are quite some alternatives for this, the stikpet library has Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes for some discussion on the differences.

### Usage

```
ts_fisher_owa(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

### Details

The formula used:

$$F_F = \frac{df_w \times SS_b}{df_b \times SSw}$$

$$df_b = k - 1$$

$$df_w = n - k$$

$$sig. = 1 - F\left(F_F, df_b, df_w\right)$$

With:

$$SS_b = \sum_{j=1}^{k} n_j \times \left(\bar{x}_j - \bar{x}\right)^2$$

$$SS_w = SS_t - SS_b$$

$$SS_t = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left(x_{i,j} - \bar{x}\right)^2$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

$$\bar{x} = \frac{\sum_{j=1}^{k} n_j \times \bar{x}_j}{n} = \frac{\sum_{j=1}^{k} \sum_{i=1}^{n_j} x_{i,j}}{n}$$

$$n = \sum_{j=1}^{k} n_j$$

Alternative format of the F-statistic equation (but the same result):

$$F_F = \frac{MS_b}{MSw}$$

*Symbols*

- $x_{i,j}$ the i-th score in category j
- $n$ the total sample size
- $n_j$ the number of scores in category j
- $k$ the number of categories
- $\bar{x}_j$ the mean of the scores in category j
- $SS_i$ the sum of squares of i (sum of squared deviation of the mean)
- $df_i$ the degrees of freedom of i
- $b$ is between = factor = treatment = model
- $w$ is within = error (the variability within the groups)

Note that the Fisher-Pitman test (Pitman, 1937a, 1937b, 1938) uses a different approach but will lead to the same result.

I'm not fully sure what the original source is for the Fisher test, but likely either of his sources from 1918, 1921, 1925 or 1935.

**Choosing a test**

The classic/Fisher one-way ANOVA assumes the data is normally distributed and that the variances in each group are the same in the population (homoscedasticity). Many have tried to cover the situations when one or both of these conditions are not met.

Delacre et al. (2019) recommend to use the Welch ANOVA instead of the classic and Brown-Forsythe versions. How2stats (2018) give a slightly different recommendation based on Tomarken and Serlin (1986). They agree that usually the Welch ANOVA is preferred of the classic version, but if the average sample size is below six to still use the Brown-Forsythe.

The researchers in the previous paragraph did not take into consideration other approaches. A few comments found on those other methods.

According to Hartung et al. (2002, p. 225) the Cochran test is the standard test in meta-analysis, but should not be used, since it is always too liberal.

Schneider and Penfield (1997) looked at the Welch, Alexander-Govern and the James test (they ignored the Brown-Forsythe since they found it to perform worse than Welch or James), and concluded: "Under variance heterogeneity, Alexander-Govern's approximation was not only comparable to the Welch test and the James second-order test but was superior, in certain instances, when coupled with the power results for those tests" (p. 285).

Cavus and Yazici (2020) compared many different tests. They showed that the Brown-Forsythe, Box correction, Cochran, Hartung-Agac-Makabi adjusted Welch, and Scott-Smith test, all do not perform well, compared to the Asiribo-Gurland correction, Alexander-Govern test, Özdemir-Kurt B2, Mehrotra modified Brown-Forsythe, and Welch.

I only came across the Johansen test in Algina et. al. (1991) and it appears to give the same results as the Welch test.

In my experience the one-way ANOVA is widely known and often discussed in textbooks. The Welch anova is gaining popularity. The Brown-Forsythe is already more obscure and some confuse it with the Brown-Forsythe test for variances. The James test and the Alexander-Govern are perhaps the least known and the Johansen even less than that (at least they were for me). So, although the

Alexander-Govern test might be preferred over the Welch test, some researchers prefer to use a more commonly used test than a more obscure version. In the end it is up to you to decide on what might be the best test, and also depending on the importance of your research you might want to investigate which test fits your situation best, rather than taking my word for it.

Besides these, there are more methods, some using simulation (bootstrapping) (see Cavus and Yazici (2020) for a few of them), others using different techniques (see Yiğit and Gökpinar (2010) for a few more methods not in here).

## Value

A dataframe with an ANOVA table showing:

| | |
|---|---|
| `variance` | which variance is shown in that row |
| `SS` | sum of squared deviations from the mean |
| `df` | degrees of freedom |
| `MS` | the mean square |
| `F` | the F-statistic value |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Algina, J., Oshima, T. C., & Tang, K. L. (1991). Robustness of Yao's, James', and Johansen's Tests under variance-covariance heteroscedasticity and nonnormality. *Journal of Educational Statistics, 16*(2), 125–139. doi:10.2307/1165116

Cavus, M., & Yazıcı, B. (2020). Testing the equality of normal distributed and independent groups' means under unequal variances by doex package. *The R Journal, 12*(2), 134. doi:10.32614/RJ-2021-008

Delacre, M., Leys, C., Mora, Y. L., & Lakens, D. (2019). Taking parametric assumptions seriously: Arguments for the use of Welch's F-test instead of the classical F-test in one-way ANOVA. *International Review of Social Psychology, 32*(1), 1–12. doi:10.5334/irsp.198

Fisher, R. A. (1921). On the "probable error" of a coefficient of correlation deduced from a small sample. *Metron, 1*, 3–32.

Hartung, J., Argaç, D., & Makambi, K. H. (2002). Small sample properties of tests on homogeneity in one-way anova and meta-analysis. *Statistical Papers, 43*(2), 197–235. doi:10.1007/s00362-002-0097-8

how2stats (Director). (2018, June 11). Welch's F-test vs Brown-Forsythe F-test: Which Should You Use and When? https://youtu.be/jteKmatBgF8

Schneider, P. J., & Penfield, D. A. (1997). Alexander and Govern's approximation: Providing an alternative to ANOVA under variance heterogeneity. *The Journal of Experimental Education, 65*(3), 271–286. doi:10.1080/00220973.1997.9943459

Tomarken, A. J., & Serlin, R. C. (1986). Comparison of ANOVA alternatives under variance heterogeneity and specific noncentrality structures. *Psychological Bulletin, 99*(1), 90–99. doi:10.1037/0033-2909.99.1.90

Yiğit, E., & Gökpinar, F. (2010). A simulation study on tests for one-way ANOVA under the unequal variance assumption. Communications, Faculty Of Science, University of Ankara, 15–34. doi:10.1501/Commua1_0000000660

---

## `ts_fligner_policello` *Fligner-Policello Test*

---

### Description

An alternative for the more famous Mann-Whitney U test. The MWU test has as an assumption that the scores in the two categories have the same shape and have unequal variances (Fong & Huang, 2019). The Fligner-Policello test does not, although the distribution around their medians should be symmetric in the population Zaiontz (n.d.).

Roughly put the assumption for this test is that the two categories have the same median in the population.

### Usage

```
ts_fligner_policello(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  ties = TRUE,
  cc = FALSE
)
```

### Arguments

| | |
|---|---|
| `catField` | A vector with the scores data |
| `ordField` | A vector with the group data |
| `categories` | optional vector with categories to use and order for the categorical field. Otherwise the first two found will be used. |
| `levels` | optional vector with the labels of the ordinal field in order. |
| `ties` | boolean to indicate the use of a ties correction. Default is TRUE |
| `cc` | boolean to indicate the use of a continuity correction. Default is FALSE |

### Details

The formula used is:

$$z = \frac{N_Y - N_X}{2 \times \sqrt{SS_X + SS_Y - M_X \times M_Y}}$$

With:

$$SS_X = \sum_{x\in X}\left(N_X - M_X\right)^2, SS_Y = \sum_{y\in Y}\left(N_Y - M_Y\right)^2$$

$$M_X = \frac{N_X}{n_x}, M_Y = \frac{N_Y}{n_y}$$

$$N_X = \sum_{x\in X} N\left(x\right), N_Y = \sum_{y\in Y} N\left(y\right)$$

$$N\left(y\right) = \sum_{x\in X} f\left(y, x\right)$$

$$N\left(x\right) = \sum_{y \in Y} f\left(x, y\right)$$

$$f\left(a, b\right) = \begin{cases} 1 & \text{if } a > b \\ 0 & \text{if } a \leq b \end{cases}$$

In case of a tie correction (Hollander et al., 2014, p. 146):

$$z = \frac{\left|N_Y - N_X\right| - 0.5}{2 \times \sqrt{SS_X + SS_Y - M_X \times M_Y}}$$

$$f\left(a, b\right) = \begin{cases} 1 & \text{if } a > b \\ 0.5 & \text{if } a = b \\ 0 & \text{if } a \leq b \end{cases}$$

*Symbols used:*

- $X$ the scores in the first category
- $Y$ the scores in the second category
- $n_i$ the number of scores in the i category

The test is described by Fligner and Policello (1981), and can also be found in Kloke and McKean (2015, p. 68)

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | test statistic |
| `p-value` | significance (p-value) |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Fligner, M. A., & Policello, G. E. (1981). Robust rank procedures for the Behrens-Fisher problem. *Journal of the American Statistical Association, 76*(373), 162–168. https://doi.org/10.1080/01621459.1981.10477623

Hollander, M., Wolfe, D. A., & Chicken, E. (2014). *Nonparametric statistical methods* (3rd ed.). John Wiley & Sons, Inc.

Kloke, J., & McKean, J. W. (2015). *Nonparametric statistical methods using R*. CRC Press, Taylor & Francis.

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
myLevels = c('Not scientific at all', 'Not too scientific', 'Pretty scientific', 'Very scientific')
ts_fligner_policello(df1[['sex']], df1[['accntsci']], levels = myLevels)
ts_fligner_policello(df1[['sex']], df1[['accntsci']], levels = myLevels, ties= FALSE, cc=TRUE)
ts_fligner_policello(df1[['sex']], df1[['accntsci']], levels = myLevels, ties= TRUE, cc=FALSE)

#Example 2: vectors
binary = c("apple", "apple", "apple", "peer", "peer", "peer", "peer")
ordinal = c(4, 3, 1, 6, 5, 7, 2)
ts_fligner_policello(binary, ordinal, categories=c("peer", "apple"))
```

---

# `ts_freeman_tukey_gof` *Freeman-Tukey Test of Goodness-of-Fit*

## Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test, but also an exact multinomial, G-test, Neyman, Mod-Log Likelihood, Cressie-Read, and Freeman-Tukey-Read test are possible.

The Freeman-Tukey attempts to make the distribution more like a normal distribution by using a square root transformation.

Lawal (1984) continued some work from Larntz (1978) and compared the modified Freeman-Tukey, G-test and the Pearson chi-square test, and concluded that for small samples the Pearson test is preferred, while for large samples either the Pearson or G-test. Making this Freeman-Tukey test perhaps somewhat redundant.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_freeman_tukey_gof(
  data,
  expCounts = NULL,
  cc = c("none", "yates", "yates2", "pearson", "williams"),
  modified = 0
)
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "yates2", "pearson", or "williams" |
| `modified` | int, optional. indicate the use of the modified version. Either 0 (default = no modification), 1 or 2 |

## Details

The formula used is (Ayinde & Abidoye, 2010, p. 21):

$$\chi_{FT}^2 = \sum_{i=1}^{k}\left(\sqrt{F_i} - \sqrt{E_i}\right)^2$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_{FT}^2, df\right)$$

With:

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

If no expected counts provided:

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

else:

$$E_i = n \times \frac{E_{p_i}}{n_p}$$

$$n_p = \sum_{i=1}^{k} E_{p_i}$$

A modified version uses another possible smoothing (Bishop, 1969, p. 284; Larntz, 1978, p.253):

$$\chi_{MFT}^2 = \sum_{i=1}^{k}\left(\sqrt{F_i} + \sqrt{F_i + 1} - \sqrt{4 \times E_i + 1}\right)^2$$

Or slightly different (Read & Cressie, 1988, p. 82):

$$\chi_{MFT}^2 = \sum_{i=1}^{k}\left(\sqrt{F_i} + \sqrt{F_i + 1} - \sqrt{4 \times (E_i + 1)}\right)^2$$

*Symbols used:*

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies
- $n_p$ the sum of all provided expected counts
- $\chi^2(\dots)$ the chi-square cumulative density function

The test is attributed to Freeman and Tukey (1950), but couldn't really find it in there. Another source often mentioned is Bishop et al. (2007)

The Yates continuity correction (cc="yates") is calculated using (Yates, 1934, p. 222):

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

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$F_i^* = \begin{cases} F_i - 0.5 & \text{if } F_i - 0.5 > E_i \\ F_i + 0.5 & \text{if } F_i + 0.5 < E_i \\ F_i & \text{else} \end{cases}$$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (cc="pearson") is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{adj}^2 = \chi_{FT}^2 \times \frac{n-1}{n}$$

The Williams correction (cc="williams") is calculated using (Williams, 1976, p. 36):

$$\chi_{adj}^2 = \frac{\chi_{FT}^2}{q}$$

With:

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

The formula is also used by McDonald (2014, p. 87)

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `test Used` | a description of the test used |

## Before, After and Alternatives

BBefore this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might an effect size measure: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

or perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Allen, A. O. (1990). *Probability, statistics, and queueing theory with computer science applications* (2nd ed.). Academic Press.

Ayinde, K., & Abidoye, A. O. (2010). Simplified Freeman-Tukey test statistics for testing probabilities in contingency tables. *Science World Journal, 2*(2), 21–27. doi:10.4314/swj.v2i2.51730

Bishop, Y. M. M., Fienberg, S. E., & Holland, P. W. (2007). *Discrete multivariate analysis*. Springer.

Bishop, Y. M. M., Fienberg, S. E., & Holland, P. W. (2007). *Discrete multivariate analysis*. Springer.

Freeman, M. F., & Tukey, J. W. (1950). Transformations Related to the angular and the square root. *The Annals of Mathematical Statistics, 21*(4), 607–611. doi:10.1214/aoms/1177729756

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

Larntz, K. (1978). Small-sample comparisons of exact levels for chi-squared goodness-of-fit statistics. *Journal of the American Statistical Association, 73*(362), 253–263. doi:10.1080/01621459.1978.10481567

Lawal, H. B. (1984). Comparisons of the X 2 , Y 2 , Freeman-Tukey and Williams's improved G 2 test statistics in small samples of one-way multinomials. *Biometrika, 71*(2), 415–418. doi:10.2307/2336263

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

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. doi:10.2307/2332518

Read, T. R. C., & Cressie, N. A. C. (1988). Goodness-of-fit statistics for discrete multivariate data. Springer-Verlag.

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

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
ts_freeman_tukey_gof(ex1)

#Example 2: Dataframe with various settings
ex2 = df1['mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_freeman_tukey_gof(ex2, expCounts=eCounts, cc="yates")
ts_freeman_tukey_gof(ex2, expCounts=eCounts, cc="pearson")
ts_freeman_tukey_gof(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_freeman_tukey_gof(ex3)
ts_freeman_tukey_gof(ex3, expCount=eCounts)
```

---

## `ts_freeman_tukey_ind` *Freeman-Tukey Test of Independence*

---

### Description

To test if two nominal variables have an association, the most commonly used test is the Pearson chi-square test of independence (Pearson, 1900). If the significance of this test is below 0.05 (or another pre-defined threshold), the two nominal variables have a significant association.

The test compares the observed counts of the cross table with the so-called expected counts. The expected values are the number of respondents you would expect if the two variables would be independent.

The Freeman-Tukey test does the same, but attempts to approximate the normal distribution with a binomial or Poisson distribution.

One problem though is that the test should only be used if not too many cells have a so-called expected count, of less than 5, and the minimum expected count is at least 1. So you will also have to check first if these conditions are met. Most often ‘not too many cells’ is fixed at no more than 20% of the cells. This is often referred to as ’Cochran conditions’, after Cochran (1954, p. 420). Note that for example Fisher (1925, p. 83) is more strict, and finds that all cells should have an expected count of at least 5 .

### Usage

```
ts_freeman_tukey_ind(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  cc = NULL,
  version = 1
)
```

### Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `cc` | optional methdod for continuity correction. Either NULL (default), "yates", "pearson", "williams". |
| `version` | optional integer to indicate which version to use. Either 1 (default), 2, or 3. |

### Details

The formula used for version 1 is (Bishop et al., 2007, p. 513):

$$T^2 = 4 \times \sum_{i=1}^{r}\sum_{j=1}^{c}\left(\sqrt{F_{i,j}} - \sqrt{E_{i,j}}\right)^2$$

The formula used for version 2 is (Lawal, 1984, p. 415):

$$T^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\left(\sqrt{F_{i,j}} + \sqrt{F_{i,j}+1} - \sqrt{4\times E_{i,j}+1}\right)^2$$

The formula used for version 3 is (Read & Cressie, 1988, p. 82):

$$T^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\left(\sqrt{F_{i,j}} + \sqrt{F_{i,j}+1} - \sqrt{4\times \left(E_{i,j}+1\right)}\right)^2$$

$$df = (r-1)\times(c-1)$$
$$sig. = 1 - \chi^2\left(T^2, df\right)$$

With:

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$
$$E_{i,j} = \frac{R_i \times C_j}{n}$$
$$R_i = \sum_{j=1}^{c} F_{i,j}$$
$$C_j = \sum_{i=1}^{r} F_{i,j}$$

*Symbols used:*

- $r$ the number of categories in the first variable (the number of rows)
- $c$ the number of categories in the second variable (the number of columns)
- $F_{i,j}$ the observed count in row i and column j
- $E_{i,j}$ the expected count in row i and column j
- $R_i$ the i-th row total
- $C_j$ the j-th column total
- $n$ the sum of all counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

The test is attributed to Freeman and Tukey (1950), but couldn't really find it in there. Ayinde and Abidoye (2010) also show the formula in more modern notation from version 1, and an another source for version 2 is Ozturk et al. (2023).

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_P^2 \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{PW}^2 = \frac{\chi_P^2}{q}$$

With:

$$q = 1 + \frac{\left(n\times\left(\sum_{i=1}^{r}\frac{1}{R_i}\right)-1\right)\times\left(n\times\left(\sum_{j=1}^{c}\frac{1}{C_j}\right)-1\right)}{6\times n\times df}$$

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `n rows` | number of categories used in first field |
| `n col.` | number of categories used in second field |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |
| `min. exp.` | the minimum expected count |
| `prop. exp. below 5` | proportion of cells with expected count less than 5 |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Ayinde, K., & Abidoye, A. O. (2010). Simplified Freeman-Tukey test statistics for testing probabilities in contingency tables. *Science World Journal, 2*(2), 21–27. doi:10.4314/swj.v2i2.51730

Bishop, Y. M. M., Fienberg, S. E., & Holland, P. W. (2007). *Discrete multivariate analysis*. Springer.

Fisher, R. A. (1925). *Statistical methods for research workers*. Oliver and Boyd.

Freeman, M. F., & Tukey, J. W. (1950). Transformations Related to the angular and the square root. *The Annals of Mathematical Statistics, 21*(4), 607–611. doi:10.1214/aoms/1177729756

Lawal, H. B. (1984). Comparisons of the X 2 , Y 2 , Freeman-Tukey and Williams's improved G 2 test statistics in small samples of one-way multinomials. *Biometrika, 71*(2), 415–418. doi:10.2307/2336263

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

Ozturk, E., Basol, M., Goksuluk, D., & Karahan, S. (2023). Performance comparison of independence tests in two-way contingency table. *REVSTAT-Statistical Journal, 21*(2), Article 2. doi:10.57805/revstat.v21i2.403

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. doi:10.2307/2332518

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

---

`ts_freeman_tukey_read` *Freeman-Tukey-Read Test of Goodness-of-Fit*

---

## Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test, but also an exact multinomial, G-test, Freeman-Tukey, Neyman, Mod-Log Likelihood and Cressie-Read test are possible.

This is actually a family (class) of tests, similar as the Cressie-Read. Weights can be chosen. the default will give the same results as the default for Cressie-Read with lambda = 0.5. Setting the weights to 4/5, 8/5, 16/15, 8/15 gives the same results as Cressie-Read with lambda = 3/2. The Pearson chi-square test is the same when setting weights to 1, 2, 1 and setting the weight simply to 4 gives the original Freeman-Tukey.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_freeman_tukey_read(
  data,
  expCounts = NULL,
  weights = c(4/3, 8/3),
  cc = c("none", "yates", "yates2", "pearson", "williams")
)
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `weights` | the weights to be used (should sum to 4) |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "pearson", or "williams" |

## Details

The formula used is (Read, 1987, p. 271):

$$FT\left(b_0, b_1, \dots, b_x\right) = \sum_{i=1}^{k}\left(\sum_{j=0}^{x} b_j \times \left(\sqrt{\frac{F_i}{E_i}}\right)^j\right) \times \left(\sqrt{F_i} - \sqrt{E_i}\right)^2$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(FT, df\right)$$

With, if no expected counts provided:

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

else:

$$E_i = n \times \frac{E_{p_i}}{n_p}$$

$$n_p = \sum_{i=1}^{k} E_{p_i}$$

The sum of the $b_i$ should be four, i.e.

$$\sum_{i=0}^{x} = 4$$

*Symbols used:*

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies
- $n_p$ the sum of all provided expected counts
- $\chi^2\left(\ldots\right)$ the chi-square cumulative density function

:

The default weights are the ones used by Read $\left(\frac{4}{3}, \frac{8}{3}\right)$, which would be the same as using a Cressie-Read power divergence with $\lambda = \frac{1}{2}$

The Yates continuity correction (cc="yates") is calculated using (Yates, 1934, p. 222):

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

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$F_i^* = \begin{cases} F_i - 0.5 & \text{if } F_i - 0.5 > E_i \\ F_i + 0.5 & \text{if } F_i + 0.5 < E_i \\ F_i & \text{else} \end{cases}$$

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

Where if $F_i^* = 0$ then $F_i^* \times ln\left(\frac{F_i^*}{E_i}\right) = 0$

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_P^2 \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{PW}^2 = \frac{\chi_P^2}{q}$$

With:

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

The formula is also used by McDonald (2014, p. 87)

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `testUsed` | a description of the test used |

## Before, After and Alternatives

BBefore this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might an effect size measure: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

or perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

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. doi:10.2307/2332518

Read, C. B. (1993). Freeman-Tukey chi-squared goodness-of-fit statistics. *Statistics & Probability Letters, 18*(4), 271–278. doi:10.1016/0167-7152(93)90015-B

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

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
ts_freeman_tukey_read(ex1)

#Example 2: pandas series with various settings
ex2 = df1['mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_freeman_tukey_read(ex2, expCounts=eCounts, cc="yates")
ts_freeman_tukey_read(ex2, expCounts=eCounts, cc="pearson")
ts_freeman_tukey_read(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
 "DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
  "DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_freeman_tukey_read(ex3)
```

---

# `ts_friedman` *Friedman Test*

## Description

A test to determine if any of the variables has a significant different average ranking than any of the others.

It is a paired-samples version of a Kruskal-Wallis test. If the p-value is below a pre-defined threshold (usually 0.05) it indicates at least one variable (column) is different than another.

## Usage

```
ts_friedman(data, levels = NULL, ties = TRUE, dist = "chi")
```

## Arguments

| | |
|---|---|
| `data` | dataframe. A column for each variable |
| `levels` | vector, optional. Indication of what the levels are in order |
| `ties` | boolean, optional. Apply a ties correction. Default is True |
| `dist` | string, optional. Distribution to use. Either "chi" (default), "f", "normal" |

## Details

The formula used in case of no ties (Friedman, 1937, p. 679):

$$\chi_F^2 = \left(\frac{12}{n \times k \times (k+1)} \times \sum_{j=1}^{k} R_j^2\right) - 3 \times n \times (k+1)$$

$$df = k - 1$$

With:

$$R_j = \sum_{i=1}^{n} r_{i,j}$$

In case a ties correction is used (Hollander & Wolfe, 1999, p. 274):

$$\chi_{Fadj}^2 = \frac{12 \times \sum_{j=1}^{k} R_j^2 - 3 \times n^2 \times (k+1)^2}{n \times (k+1) - \frac{\left(\sum t_{i,j}^3\right) - k}{k-1}}$$

The ties correction used by IBM SPSS (2021, p. 811) will give the same result:

$$\chi_{Fadj}^2 = \frac{\chi_F^2}{1 - \frac{\sum t_{i,j}^3 - t_{i,j}}{n \times (k^3 - k)}}$$

The function uses more of a one-way ANOVA approach in case of ties, but then on the ranks. It leads to the same result:

$$\chi_{Fadj}^2 = \frac{n \times \sum_{j=1}^{k} \left(\bar{r}_j - \bar{r}\right)^2}{\left(\frac{\sum_{j=1}^{k} \sum_{i=1}^{n} \left(r_{i,j} - \bar{r}\right)^2}{n \times (k-1)}\right)}$$

With:

$$\bar{r}_j = \frac{R_j}{n}$$

$$\bar{r} = \frac{\sum_{j=1}^{k} R_j}{n \times k} = \frac{n \times (k+1)}{2}$$

The significance is then determined using:

$$sig. = 1 - \chi^2\left(\chi_F^2, df\right)$$

A normal distribution approximation was proposed by Friedman (1937, p. 695; 1939, p. 109):

$$z_F = \frac{\chi_F^2 - (k-1)}{\sqrt{2 \times \frac{n-1}{n} \times (k-1)}}$$

$$sig. = 2 \times (1 - \Phi(|z_F|))$$

And an F distribution by Iman and Davenport (1980, p. 573):

$$F_F = \frac{(n-1) \times \chi_F^2}{n \times (k-1) - \chi_F^2}$$

$$df_1 = k - 1$$

$$df_2 = (k-1) \times (n-1)$$

$$sig. = 1 - F\left(F_F, df_1, df_2\right)$$

Some might refer to Conover for this F-distribution, but in Conover (1980, p. 300) it seems Conover credits Iman and Davenport himself.

*Symbols Used*

- $n$, the number of cases
- $k$, the number of variables
- $r_{i,j}$, the rank of case i, in variable j. The ranks are determined for each case.
- $t_{i,j}$, the frequency of unique rank j, in case i. For each row the frequencies of each rank is determined in the calculations.

### Value

res : dataframe with the following columns

| | |
|---|---|
| `n` | sample size |
| `statistic` | test statistic used |
| `df, df1, df2` | degrees of freedom (if applicable) |
| `p-value` | the p-value (significance) |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Conover, W. J. (1980). *Practical nonparametric statistics* (2nd ed.). Wiley.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association, 32*(200), 675–701. doi:10.2307/2279372

Friedman, M. (1939). A correction. *Journal of the American Statistical Association, 34*(205), 109–109. doi:10.1080/01621459.1939.10502372

Hollander, M., & Wolfe, D. A. (1999). *Nonparametric statistical methods* (2nd ed.). Wiley.

IBM. (2021). IBM SPSS Statistics Algorithms. IBM.

Iman, R., & Davenport, J. (1980). Approximations of the critical region of the Friedman statistic. *Communications in Statistics-Theory and Methods, 9*, 571–595.

---

## `ts_g_gof` *G (Likelihood Ratio) Test of Goodness-of-Fit*

### Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test, but also an exact multinomial, Freeman-Tukey, Neyman, Mod-Log Likelihood and Cressie-Read test are possible.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_g_gof(
  data,
  expCounts = NULL,
  cc = c("none", "yates", "yates2", "pearson", "williams")
)
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "pearson", or "williams" |

## Details

The formula used (Wilks, 1938, p. 62):

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

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(G, df\right)$$

With:

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

If no expected counts provided:

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

else:

$$E_i = n \times \frac{E_{p_i}}{n_p}$$

$$n_p = \sum_{i=1}^{k} E_{p_i}$$

*Symbols used*:

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies
- $n_p$ the sum of all provided expected counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

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)

The Yates continuity correction (cc="yates") is calculated using (Yates, 1934, p. 222):

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

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$F_i^* = \begin{cases} F_i - 0.5 & \text{if } F_i - 0.5 > E_i \\ F_i + 0.5 & \text{if } F_i + 0.5 < E_i \\ F_i & \text{else} \end{cases}$$

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

Where if $F_i^* = 0$ then $F_i^* \times ln\left(\frac{F_i^*}{E_i}\right) = 0$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (cc="pearson") is calculated using (E.S. Pearson, 1947, p. 157):

$$G_P = G \times \frac{n-1}{n}$$

The Williams correction (cc="williams") is calculated using (Williams, 1976, p. 36):

$$G_W = \frac{G}{q}$$

With:

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

The formula is also used by McDonald (2014, p. 87)

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `test used` | a description of the test used |

## Before, After and Alternatives

Before this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might an effect size measure: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

or perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

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

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

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

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. doi: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. doi: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. doi:10.1214/aoms/1177732360

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
ts_g_gof(ex1)

#Example 2: dataframe with various settings
ex2 = df1['mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_g_gof(ex2, expCounts=eCounts, cc="yates")
ts_g_gof(ex2, expCounts=eCounts, cc="pearson")
ts_g_gof(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
```

```
ts_g_gof(ex3)
```

---

## `ts_g_ind` *G (Likelihood Ratio / Wilks) Test of Independence*

### Description

This test is similar as a Pearson Chi-Square test of independence, but approaches it from a likelihood-ratio approach (see Monica, 2015).

If the significance of this test is below 0.05 (or another pre-defined threshold), the two nominal variables have a significant association.

The test compares the observed counts of the cross table with the so-called expected counts. The expected values are the number of respondents you would expect if the two variables would be independent.See the Pearson Chi-Square test of independence for more details on expected counts.

One problem though is that the test should only be used if not too many cells have a so-called expected count, of less than 5, and the minimum expected count is at least 1. So you will also have to check first if these conditions are met. Most often ‘not too many cells’ is fixed at no more than 20% of the cells. This is often referred to as ’Cochran conditions’, after Cochran (1954, p. 420). Note that for example Fisher (1925, p. 83) is more strict, and finds that all cells should have an expected count of at least 5 .

### Usage

```
ts_g_ind(field1, field2, categories1 = NULL, categories2 = NULL, cc = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `cc` | optional methdod for continuity correction. Either NULL (default), "yates", "pearson", "williams". |

### Details

The formula used (Wilks, 1938, p. 62):

$$G = 2 \times \sum_{i=1}^{r}\sum_{j=1}^{c}\left(F_{i,j} \times ln\left(\frac{F_{i,j}}{E_{i,j}}\right)\right)$$

$$df = (r-1) \times (c-1)$$

$$sig. = 1 - \chi^2\left(G, df\right)$$

With:

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

Use instead of $F_{i,j}$ the adjusted version defined by:

$$F_{i,j}^{*} = \begin{cases} F_{i,j} - 0.5 & \text{if } F_{i,j} > E_{i,j} \\ F_{i,j} & \text{if } F_{i,j} = E_{i,j} \\ F_{i,j} + 0.5 & \text{if } F_{i,j} < E_{i,j} \end{cases}$$

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^{2} = G^{2} \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using:

$$\chi_{PW}^{2} = \frac{G}{q}$$

With:

$$q = 1 + \frac{\left(n \times \left(\sum_{i=1}^{r} \frac{1}{R_i}\right) - 1\right) \times \left(n \times \left(\sum_{j=1}^{c} \frac{1}{C_j}\right) - 1\right)}{6 \times n \times df}$$

The formula is probably from Williams (1976, p. 36) but the one shown here is taken from McDonald (1976, p. 36).

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `n rows` | number of categories used in first field |
| `n col.` | number of categories used in second field |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |
| `min. exp.` | the minimum expected count |
| `prop. exp. below 5` | proportion of cells with expected count less than 5 |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cochran, W. G. (1954). Some methods for strengthening the common $\chi^2$ tests. *Biometrics, 10*(4), 417. doi:10.2307/3001616

Fisher, R. A. (1925). *Statistical methods for research workers*. Oliver and Boyd.

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

Monica, gung-R. (2015, April 2). Answer to "Why do my p-values differ between logistic regression output, chi-squared test, and the confidence interval for the OR?" Cross Validated. https://stats.stackexchange.com/a/

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

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. https://doi.org/10.2307/2983604

---

# `ts_ham_owa` *Hartung-Argaç-Makambi Test*

## Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

This test is a modification of the Welch one-way ANOVA.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

## Usage

```
ts_ham_owa(nomField, scaleField, categories = NULL, version = c(1, 2))
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |
| `version` | the phi method calculation to use (see details) |

## Details

The formula used (Hartung et al., 2002, p. 206):

$$W = \frac{\frac{1}{k-1}\times\sum_{j=1}^{k} w_j^*\times\left(\bar{x}_j - \bar{y}_w^*\right)^2}{1 + \frac{2\times(k-2)}{k^2-1}\times\lambda^*}$$

$$df_1 = k - 1$$

$$df_2 = \frac{k^2-1}{3\times\lambda^*}$$

$$sig. = 1 - F\left(W, df_1, df_2\right)$$

With:

$$\bar{y}_w^* = \sum_{j=1}^{k} h_j^*\times\bar{x}_j$$

$$h_j^* = \frac{w_j^*}{w^*}$$

$$w_j^* = \frac{n_j}{s_j^2}\times\frac{1}{\phi_j}$$

$$w^* = \sum_{j=1}^{k} w_j^*$$

$$\phi_j = \frac{n_j+2}{n_j+1}$$

$$\bar{x}_j = \frac{\sum_{j=1}^{n_j} x_{i,j}}{n_j}$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\lambda^* = \sum_{j=1}^{k}\frac{\left(1 - h_j^*\right)^2}{n_j - 1}$$

*Symbols used:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $x_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $w_j^*$ the modified weight for category j
- $h_j^*$ the adjusted modified weight for category j
- $df_i$ the i-th degrees of freedom

Note that the numerator in $W$ is the same as the Cochran test statistic.

Cavis and Yazici (2020, p. 6) uses $\phi_j = \frac{n_j-1}{n_j-3}$. However the original article states that these are unbalanced weights of the Welch test and in their experience, using these makes the test too conservative. In the original article they find from their simulation experience that using (n_j+2)/(n_j+1) gives reliable results for small sample sizes, and a large number of populations (Hartung et al. p. 207).

By setting 'version=2' the same version for $\phi$ as in the Doex library will be used

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the test statistic (F value) |
| `df1` | the degrees of freedom 1 |
| `df2` | the degrees of freedom 2 |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cavus, M., & Yazıcı, B. (2020). Testing the equality of normal distributed and independent groups' means under unequal variances by doex package. *The R Journal, 12*(2), 134. https://doi.org/10.32614/RJ-2021-008

Hartung, J., Argaç, D., & Makambi, K. H. (2002). Small sample properties of tests on homogeneity in one-way anova and meta-analysis. *Statistical Papers, 43*(2), 197–235. https://doi.org/10.1007/s00362-002-0097-8

---

# `ts_james_owa` *James One-Way Test*

## Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

James (1951) proposed three tests, one for large group sizes, a 'first order test', and a 'second order test'. The later two a significance level ($\alpha$) is chosen and a critical value is then calculated based on a modification of the chi-square distribution.

The James test statistic value J is the same as the test statistic in Cochran's test, calculated slightly different, but will lead to the same result.

Schneider and Penfield (1997) looked at the Welch, Alexander-Govern and the James test (they ignored the Brown-Forsythe since they found it to perform worse than Welch or James), and concluded: "Under variance heterogeneity, Alexander-Govern's approximation was not only comparable to the Welch test and the James second-order test but was superior, in certain instances, when coupled with the power results for those tests" (p. 285).

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

## Usage

```
ts_james_owa(
  nomField,
  scaleField,
  categories = NULL,
  order = c(0, 1, 2),
  ddof = 2
)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |
| `order` | the order of the James test to perform (see details) |
| `ddof` | int, optional. Offset for degrees of freedom. Default is 2. |

## Details

The formula use for the test statistic (James, 1951, p. 324):

$$J = \sum_{j=1}^{k} w_j \times \bar{x}_j^2 - \frac{\left(\sum_{s=1}^{k} w_s \times \bar{x}_s\right)^2}{w} = \chi_{Cochran}^2$$

With:

$$w_j = \frac{n_j}{s_j^2}$$

$$h_j = \frac{w_j}{w}$$

$$w = \sum_{j=1}^{k} w_j$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

#' *Symbols:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $x_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $w_j$ the weight for category j
- $h_j$ the adjusted weight for category j

For large group size (order=0) the same result as the Cochran test (James, 1951, p. 324):

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(J, df\right)$$

The first order James test (order=1) is done using (James, 1951, p. 324):

$$J_{crit} = \chi^2_{crit} \times \left(1 + \frac{3 \times \chi^2_{crit} + k + 1}{2 \times \left(k^2 - 1\right)} \times \lambda\right)$$

With:

$$\lambda = \sum_{j=1}^{k} \frac{\left(1 - h_j\right)^2}{v_j}$$

$$\chi^2_{crit} = Q\left(\chi^2\left(1 - \alpha, df\right)\right)$$

$$v_j = n_j - 1$$

*Symbols used:*

- $\chi^2_{crit}$ the critical chi-square value at alpha level

The second order James test (order=2) is done using (James)

$$J_{crit} = C + \frac{1}{2} \times (3 \times \chi_4 + \chi_2) \times \lambda + \frac{1}{16} \times (3 \times \chi_4 + \chi_2)^2 \times \Bigg(1 - \frac{k-3}{C}) \times \lambda^2 + \frac{1}{2} \times (3 \times \chi_4 + \chi_2) \times \Bigg(\big(8 \times R_{23} - 1$$

With:

$$\lambda_2 = \sum_{j=1}^{k} \frac{\left(1 - h_j\right)^2}{v_j^*}$$

$$v_j^* = n_j - 2$$

$$\chi_{2 \times s} = \frac{\left(\chi^2_{crit}\right)^s}{\prod_{i=1}^{s}\left(k + 2 \times i - 3\right)}$$

$$R_{xy} = \sum_{j=1}^{k} \frac{h_j^y}{\left(v_j^*\right)^x}$$

This function will do an iterative search to find the approximate p-value

The use of $v_j^* = n_j - 2$ for the James order 2 test is based on James (1951, p. 328) which can also be found in Deshon and Alexander (1994, p. 331).

However, others use $v_j^* = n_j - 1$ for example Myers (1998, p. 209) and Cribbie et al. (2012, p. 62) By setting 'ddof' this alternative version will be used.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the J-statistic from the test |
| `Jcrit` | critical J value |
| `df` | the degrees of freedom |
| `pValue` | the significance (p-value) |
| `test` | description of test used |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cribbie, R. A., Fiksenbaum, L., Keselman, H. J., & Wilcox, R. R. (2012). Effect of non-normality on test statistics for one-way independent groups designs: Effects of non-normality on test statistics. *British Journal of Mathematical and Statistical Psychology, 65*(1), 56–73. doi:10.1111/j.2044-8317.2011.02014.x

Deshon, R. P., & Alexander, R. A. (1994). A generalization of James's second-order approximation to the test for regression slope equality. *Educational and Psychological Measurement, 54*(2), 328–335. doi:10.1177/0013164494054002007

James, G. S. (1951). The comparison of several groups of observations when the ratios of the population variances are unknown. *Biometrika, 38*(3–4), 324–329. doi:10.1093/biomet/38.3-4.324

Myers, L. (1998). Comparability of the james' second-order approximation test and the alexander and govern A statistic for non-normal heteroscedastic data. *Journal of Statistical Computation and Simulation, 60*(3), 207–222. doi:10.1080/00949659808811888

Schneider, P. J., & Penfield, D. A. (1997). Alexander and Govern's approximation: Providing an alternative to ANOVA under variance heterogeneity. *The Journal of Experimental Education, 65*(3), 271–286. doi:10.1080/00220973.1997.9943459

---

## `ts_kruskal_wallis` *Kruskal-Wallis H Test*

### Description

This test is an extension of the Mann-Whitney U test (see ts_mann_whitney()) to more than two categories. It is also seen as the non-parametric version of the one-way ANOVA (see ts_fisher_owa()).

The test can indicate if any of the scores in one or more categories, has a significant different mean rank than one or more of the other categories. More strickly the null hypothesis is that the probability of a randomly selected case having a score greater than a random score from the other category is 50% (Divine et al., p. 286).

Alternative there is a Mood Median test (see ts_mood_median()).

To pin-point which category or categories differ significantly, a post-hoc analysis could be used.

### Usage

```
ts_kruskal_wallis(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  method = c("chi2", "kw-gamma", "kw-gamma-chi2", "kw-beta", "kw-beta-f",
    "wallace-I-beta", "wallace-II-beta", "wallace-III-beta", "wallace-I-f",
    "wallace-II-f", "wallace-III-f", "iman"),
  tiescorr = c(TRUE, FALSE)
)
```

## Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `method` | string, optional. the specific variation of the test to use. Default is "chi2". Options are: : "chi2", "kw-gamma", "kw-gamma-chi2", "kw-beta", "kw-beta-f", "wallace-f1", "wallace-f2", "wallace-f3", "wallace-beta1", "wallace-beta2", "wallace-beta3", "ids" |
| `tiescorr` | boolean, optional. use of a ties correction. Default is True. |

## Details

### The H value

The formula used is (Kruskal & Wallis, 1952, p. 586):

$$H = \frac{12}{n\times\left(n+1\right)}\times\sum_{i=1}^k \frac{R_i^2}{n_i}-3\times\left(n+1\right)$$

With:

$$R_i = \sum_{j=1}^{n_i} r_{i,j}$$

The ties correction (Kruskal & Wallis, 1952, p. 586):

$$H_{adj} = \frac{H}{1-\frac{\sum T}{n^3-n}}$$

With:

$$T_j = t_j^3 - t_j$$

Or alternatively:

$$H_{adj} = \left(n-1\right)\times\frac{\sum_{i=1}^k n_i\left(\bar{r}_i-\bar{r}\right)^2}{\sum_{i=1}^k\sum_{j=1}^{n_i}\left(r_{i,j}-\bar{r}\right)^2}$$

With:

$$\bar{r}_i = \frac{R_i}{n_i}$$

$$\bar{r} = \frac{\sum_{i=1}^k \bar{r}_i}{\sum_{i=1}^k n_i}$$

### The Test

*"chi2", Kruskal-Wallis Chi-Square Approximation*

$$sig. \approx 1 - \chi^2\left(H, df\right)$$

$$df = k - 1$$

*"kw-gamma", Kruskal-Wallis incomplete gamma approximation* (Kruskal & Wallis, 1952, p. 609)

$$sig. \approx 1 - \gamma\left(H, \alpha, \beta\right)$$

$$\alpha = \frac{\mu^2}{\sigma^2}$$

$$\beta = \frac{\sigma^2}{\mu}$$

$$\mu = k - 1$$

$$\sigma^2 = 2 \times (k-1) - \frac{2 \times \left(k \times k^2 - 6 \times k + n \times \left(2 \times k^2 - 6 \times k + 1\right)\right)}{5 \times n \times (n+1)} - \frac{6}{5} \times \sum_{i=1}^{k} \frac{1}{n_i}$$

*"kw-gamma-chi2", Kruskal-Wallis Chi-square approximation of gamma approximation*

$$sig. = 1 - \chi^2\left(\chi_a^2, df\right)$$

$$\chi_a^2 = \frac{2 \times \mu}{\sigma^2} \times H$$

$$df = 2 \times \frac{\mu^2}{\sigma^2}$$

*"kw-beta", Kruskal-Wallis incomplete Beta distribution approximation* (Kruskal & Wallis, 1952, p. 609)

$$sig. = 1 - \beta\left(\frac{H}{M}, \alpha, \beta\right)$$

$$M = \frac{n^3 - \sum_{i=1}^{k} n_i^3}{n \times (n+1)}$$

$$\alpha = df_1 \times \frac{1}{2}$$

$$\beta = df_2 \times \frac{1}{2}$$

$$df_1 = \mu \times \frac{\mu \times (M - \mu) - \sigma^2}{\frac{1}{2} \times M \times \sigma^2}$$

$$df_2 = df_1 \times \frac{M - \mu}{\mu}$$

*"kw-beta-f", F-approximation of the Kruskal-Wallis incomplete Beta distribution approximation* (Kruskal & Wallis, 1952, p. 610)

$$sig. \approx 1 - F\left(F_\alpha, df_1, df_2\right)$$

$$F_\alpha = \frac{H \times (M - \mu)}{\mu \times (M - H)}$$

*Wallace F distribution approximations* (Wallace, 1959, p. 226)

$$sig. = 1 - F\left(F_2, df_1^i, df_2^i\right)$$

With:

$$F_2 = \frac{(n-k) \times H}{(k-1) \times (n-1-H)}$$

$$df_1^i = (k-1) \times d_i$$

$$df_2^i = (n-k) \times d_i$$

*Wallace Beta distribution approximations* (Wallace, 1959, p. 226)

$$sig. \approx 1 - \beta\left(B_2, \alpha, \beta\right)$$

$$B_2 = \frac{H}{n-1}$$

$$\alpha = df_1 \times \frac{1}{2}$$

$$\beta = df_2 \times \frac{1}{2}$$

*"wallace-f1"* and *"wallace-b1"*

$$d_i = \frac{(n-k) \times (k-1) - \sigma^2}{\frac{1}{2} \times (n-1) \times \sigma^2}$$

*"wallace-f2"* and *"wallace-b2"*

$$d_i = 1 - \frac{6 \times (n+1)}{5 \times (n-1) \times (n+1.2)}$$

*"wallace-f3"* and *"wallace-b3"*

$$d_i = 1$$

*"ids", Iman-Davenport Satterwaite approximation* (Iman & Davenport, 1976, p. 1338)

$$sig. = 1 - F\left(F_2, df_1, df_2\right)$$

With:

$$df_1 = k - 1$$

$$df_2 = \frac{\left(\sum_{i=1}^{k}\left(n_i - 1\right) \times v_i\right)^2}{\sum_{i=1}^{k}\frac{\left(\left(n_i-1\right) \times v_i\right)^2}{n_i - 1}}$$

$$v_i = \frac{\sum_{j=1}^{n_i}\left(r_{i,j} - \bar{r}_i\right)^2}{n_i - 1}$$

$$\bar{r}_i = \frac{\sum_{j=1}^{n_i} r_{i,j}}{n_i}$$

*Symbols used:*

- $k$, the number of categories
- $t_j$, the frequency of the j-th unique rank.
- $n$, the total sample size
- $n_i$, the number of scores in category i
- $r_{i,j}$, the rank of the j-th score in category i
- $R_i$, the sum of the ranks in category i
- $\bar{r}_i$, the average of the ranks in category i
- $\bar{r}$, the average of all ranks
- $\chi^2\left(\dots\right)$, the cumulative distribution function of the chi-square distribution.
- $F\left(\dots\right)$, the cumulative distribution function of the F distribution.
- $\beta\left(\dots\right)$, the cumulative distribution function of the beta distribution.

I have not been able to find an exact distribution for H in R. A good starting point might be Choi et al. (2003) and let me know if you manage.

### Value

Returns a dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `H` | the H value |

Depending on the test used additional items might be added. In case of a chi-square approximation

| | |
|---|---|
| `statistic` | if not the same as H, the chi-square value used |
| `df` | the degrees of freedom |

In case of a gamma, or beta approximation

| | |
|---|---|
| `statistic` | the test statistic used |
| `alpha` | the alpha value used |
| `beta` | the beta value used |

In case of a F approximation

| | |
|---|---|
| `statistic` | the test statistic used |
| `df1` | the first degrees of freedom |
| `df2` | the second degrees of freedom |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Iman, R. L., & Davenport, J. M. (1976). New approximations to the exact distribution of the kruskal-wallis test statistic. *Communications in Statistics - Theory and Methods, 5*(14), 1335–1348. doi:10.1080/03610927608827446

Kruskal, W. H., & Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association, 47*(260), 583–621. doi:10.1080/01621459.1952.10483441

Wallace, D. L. (1959). Simplified beta-approximations to the Kruskal-Wallis H test. *Journal of the American Statistical Association, 54*(285), 225. doi:10.2307/2282148

---

## `ts_mann_whitney` *Mann-Whitney U Test*

### Description

The Mann-Whitney U and Wilcoxon Rank Sum test are the same. Mann and Whitney simply expanded on the ideas from Wilcoxon.

The test will compare the distribution of ranks between two categories. The assumption is that the two categories have the same mean rank (which often is stated simplified as having the same median in the population).

## Usage

```
ts_mann_whitney(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  method = "exact",
  cc = TRUE
)
```

## Arguments

| | |
|---|---|
| `catField` | A vector or dataframe with the group data |
| `ordField` | A vector or dataframe with the scores data |
| `categories` | : optional list with the two categories to use from catField. If not set the first two found will be used |
| `levels` | optional list with the scores in order |
| `method` | c("exact", "appr") exact method or normal approximation |
| `cc` | boolean to indicate the use of a continuity correction |

## Details

The formula used is (Mann & Whitney, 1947, p. 51):

$$U_i = R_i - \frac{n_i \times (n_i + 1)}{2}$$

With:

$$R_i = \sum_{j=1}^{n_i} r_{i,j}$$

For an approximation the following is used:

$$sig. = 2 \times (1 - Z(z))$$

With:

$$z = \frac{U_i - \frac{n_1 \times n_2}{2}}{SE}$$

$$SE = \sqrt{\frac{n_1 \times n_2}{n \times (n-1)} \times \left(\frac{n^3 - n}{12} - \sum_i T_i\right)}$$

$$T_i = \frac{t_i^3 - t_i}{12}$$

$$n = n_1 + n_2$$

If a continuity correction is used the z-value is calculated using:

$$z_{cc} = z - \frac{0.5}{SE}$$

*Symbols used:*

- $n_i$ the sample size of category i
- $n$ the total sample size
- $r_{i,j}$ the j-th rank of category i

The ties correction ($T$) can be found in Lehmann and D'Abrera (1975, p. 20)

For the exact distribution the Mann-Whitney-Wilcoxon distribution is used, from the **pwilcox()** function from R.

Wilcoxon (1945) had developed this test earlier for the case when both categories have the same sample size, and Mann and Whitney expanded on this.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `U1` | the Mann-Whitney U score of the first category |
| `U2` | the Mann-Whitney U score of the second category |
| `statistic` | test statistic |
| `pValue` | significance (p-value) |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Lehmann, E. L., & D'Abrera, H. J. M. (1975). *Nonparametrics: Statistical methods based on ranks*. Holden-Day.

Mann, H. B., & Whitney, D. R. (1947). On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics, 18*(1), 50–60. https://doi.org/10.1214/aoms/1177730491

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin, 1*(6), 80. https://doi.org/10.2307/3001968

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
myLevels = c('Not scientific at all', 'Not too scientific', 'Pretty scientific', 'Very scientific')
ts_mann_whitney(df1['sex'], df1['accntsci'], levels=myLevels)

#Example 2: vectors
binary = c("apple", "apple", "apple", "peer", "peer", "peer", "peer")
ordinal = c(4, 3, 1, 6, 5, 7, 2)
ts_mann_whitney(binary, ordinal, categories=c("peer", "apple"))
```

## `ts_mcnemar_bowker` *(McNemar-)Bowker Test*

### Description

The Bowker test (Bowker, 1948) is an extension of the McNemar (1947) test, which was only for 2x2 tables.

It tests if there is a change in symmetric opinion changes. It assumes there is no change, and if the p-value is below a pre-set threshold (usually 0.05) this assumption is rejected.

### Usage

```
ts_mcnemar_bowker(field1, field2, categories = NULL, cc = FALSE)
```

### Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |
| `cc` | boolean, optional, use of a continuity correction (default is False) |

### Details

The formula used is (Bowker, 1948, p. 573):

$$\chi_{B}^2 = \sum_{i=1}^{r-1}\sum_{j=i+1}^{c}\frac{\left(F_{i,j}-F_{j,i}\right)^2}{F_{i,j}+F_{j,i}}$$

$$df = \frac{r\times\left(r-1\right)}{2} = \frac{c\times\left(c-1\right)}{2}$$

$$sig. = 1 - \chi^2\left(\chi_{B}\right)$$

*Symbols used*

- $r$ is the number of rows (categories in the first variable)
- $c$ is the number of columns (categories in the second variable)
- $n$ is the total number of scores
- $F_{i,j}$ is the frequency (count) of scores equal to the i-th category in the first variable, and the j-th category in the second.

### Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the chi-squared value |
| `df` | the degrees of freedom used in the test |
| `p-value` | the significance (p-value) |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Bowker, A. H. (1948). A test for symmetry in contingency tables. *Journal of the American Statistical Association, 43*(244), 572–574. doi:10.2307/2280710

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, 12*(2), 153–157. doi:10.1007/BF02295996

---

## `ts_mehrotra_owa` *Mehrotra Test*

---

### Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

Mehrotra (1997) modified the calculation for the first degrees of freedom in the Brown-Forsythe test for means, all other values are the same.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

### Usage

```
ts_mehrotra_owa(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

### Details

The formula used (Mehrotra, 1997, p. 11141):

$$F_M = \frac{\sum_{j=1}^{k} n_j \times \left(\bar{x}_j - \bar{x}\right)^2}{\sum_{j=1}^{k} \left(1 - \frac{n_j}{n}\right) \times s_j^2}$$

$$df_1 = \frac{\left(\sum_{j=1}^{k} s_j^2 - \frac{n_j \times s_j^2}{n}\right)^2}{\sum_{j=1}^{k} s_j^4 + \left(\frac{\sum_{j=1}^{k} n_j \times s_j^2}{n}\right)^2 - 2 \times \frac{\sum_{j=1}^{k} n_j \times s_j^4}{n}}$$

$$df_2 = \frac{\left(\sum_{j=1}^{k}\left(1-\frac{n_j}{n}\right)\times s_j^2\right)^2}{\sum_{j=1}^{k}\frac{\left(1-\frac{n_j}{n}\right)\times s_j^4}{n_j-1}}$$

$$sig. = 1 - F\left(F_{BF}, df_1, df_2\right)$$

With:

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j}-\bar{x}_j\right)^2}{n_j-1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

$$\bar{x} = \frac{\sum_{i=1}^{k} n_j \times \bar{x}_j}{n}$$

$$n = \sum_{j=1}^{k} n_j$$

*Symbols:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n$ the total sample size
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $df$ the degrees of freedom
- $F\left(\dots,\dots,\dots\right)$ the cumulative distribution function of the F distribution.

The same as the Brown-Forsythe test for means, except for $df_1$.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the test statistic (F value) |
| `df1` | the degrees of freedom 1 |
| `df2` | the degrees of freedom 2 |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Mehrotra, D. V. (1997). Improving the Brown-Forsythe solution to the generalized Behrens-Fisher problem. *Communications in Statistics - Simulation and Computation, 26*(3), 1139–1145. doi:10.1080/03610919708813

---

`ts_mod_log_likelihood_gof` 
*Mod-Log Likelihood Test of Goodness-of-Fit*

---

## Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test, but also an exact multinomial, G-test, Freeman-Tukey, Neyman, Cressie-Read, and Freeman-Tukey-Read test are possible.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_mod_log_likelihood_gof(
  data,
  expCounts = NULL,
  cc = c("none", "yates", "yates2", "pearson", "williams")
)
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "pearson", or "williams" |

## Details

The formula used (Cressie & Read, 1984, p. 441):

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

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_{MLR}^2, df\right)$$

With:

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

If no expected counts provided:

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

else:

$$E_i = n\times\frac{E_{p_i}}{n_p}$$

$$n_p = \sum_{i=1}^{k} E_{p_i}$$

*Symbols used*:

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies
- $n_p$ the sum of all provided expected counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

Cressie and Read (1984) is not the original source, but the source where I found the formula.

The Yates continuity correction (cc="yates") is calculated using (Yates, 1934, p. 222):

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

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$F_i^* = \begin{cases} F_i - 0.5 & \text{if } F_i - 0.5 > E_i \\ F_i + 0.5 & \text{if } F_i + 0.5 < E_i \\ F_i & \text{else} \end{cases}$$

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

Where if $E_i = 0$ then $E_i \times ln\left(\frac{E_i}{F_i^*}\right) = 0$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (cc="pearson") is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi^2_{MLREP} = \chi^2_{MLR} \times \frac{n-1}{n}$$

The Williams correction (cc="williams") is calculated using (Williams, 1976, p. 36):

$$\chi^2_{MLRW} = \frac{\chi^2_{MLR}}{q}$$

With:

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

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `test used` | a description of the test used |

## Before, After and Alternatives

BBefore this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might an effect size measure: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

or perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cressie, N., & Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society: Series B (Methodological), 46*(3), 440–464. doi:10.1111/j.2517-6161.1984.tb01318.x

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

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. doi:10.2307/2332518

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
ts_mod_log_likelihood_gof(ex1)

#Example 2: dataframe with various settings
ex2 = df1['mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_mod_log_likelihood_gof(ex2, expCounts=eCounts, cc="yates")
ts_mod_log_likelihood_gof(ex2, expCounts=eCounts, cc="pearson")
ts_mod_log_likelihood_gof(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED",
"SEPARATED", "DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_mod_log_likelihood_gof(ex3)
```

---

# `ts_mod_log_likelihood_ind` *Mod-Log Likelihood Test of Independence*

---

## Description

Mod-Log Likelihood Test of Independence

## Usage

```
ts_mod_log_likelihood_ind(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  cc = NULL
)
```

## Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `cc` | optional methdod for continuity correction. Either NULL (default), "yates", "pearson", "williams". |

## Details

The formula used (Cressie & Read, 1984, p. 441):

$$MG = 2 \times \sum_{i=1}^{r}\sum_{j=1}^{c}\left(E_{i,j} \times ln\left(\frac{E_{i,j}}{F_{i,j}}\right)\right)$$

$$df = (r-1) \times (c-1)$$

$$sig. = 1 - \chi^2\left(MG, df\right)$$

With:

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

Cressie and Read (1984) is not the original source, but the source where I found the formula.

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

Use instead of $F_{i,j}$ the adjusted version defined by:

$$F_{i,j}^* = \begin{cases} F_{i,j} - 0.5 & \text{if } F_{i,j} > E_{i,j} \\ F_{i,j} & \text{if } F_{i,j} = E_{i,j} \\ F_{i,j} + 0.5 & \text{if } F_{i,j} < E_{i,j} \end{cases}$$

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_P^2 \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{PW}^2 = \frac{\chi_P^2}{q}$$

With:

$$q = 1 + \frac{\left(n \times \left(\sum_{i=1}^{r} \frac{1}{R_i}\right) - 1\right) \times \left(n \times \left(\sum_{j=1}^{c} \frac{1}{C_j}\right) - 1\right)}{6 \times n \times df}$$

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `n rows` | number of categories used in first field |
| `n col.` | number of categories used in second field |
| `statistic` | the test statistic (chi-square value) |

| | |
|---|---|
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |
| `min. exp.` | the minimum expected count |
| `prop. exp. below 5` | proportion of cells with expected count less than 5 |
| `test` | description of the test used |

## Author(s)

P. Stikker

Please visit: https://PeterStatistics.com

YouTube channel: https://www.youtube.com/stikpet

## References

Cressie, N., & Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society: Series B (Methodological), 46*(3), 440–464. https://doi.org/10.1111/j.2517-6161.1984.tb01318.x

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

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. https://doi.org/10.2307/2983604

## Examples

```
nom1 <- c("female", "female","female","female","female","female","female",
"female", "female","female","female", "male", "male", "male", "male", "male",
"male", "male", "male", "male", "male", "male", "male", "male", "male", "male",
"male","male", "male", "male", "male", "male", "male", "male", "male", "male",
"male", "male", "male", "male", "male", "male")
nom2 <- c("nl", "nl","nl","nl","nl","nl","nl","nl", "other", "other", "other",
"nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl","nl",
"other", "other", "other", "other", "other", "other", "other", "other", "other",
"other", "other", "other", "other", "other", "other")
ts_g_ind(nom1, nom2)
ts_g_ind(nom1, nom2, cc="yates")
ts_g_ind(nom1, nom2, cc="pearson")
ts_g_ind(nom1, nom2, cc="williams")
```

---

## `ts_mood_median` *Mood Median Test*

## Description

This test looks if the median from different categories would be the same in the population. If not, at least one is different then at least one other category. A Kruskal-Wallis test (see ts_kruksal_wallis()) is very similar but checks the average ranks instead of median.

The test only looks at the number of scores above the overall median and those that are equal or below. A cross table is made with each category and the numbers below and above the overall median. From this table a test of independence can be used.

## Usage

```
ts_mood_median(
  catField,
  ordField,
  categories = NULL,
  levels = NULL,
  test = "pearson",
  cc = c(NULL, "yates", "pearson", "williams"),
  lambd = 2/3
)
```

## Arguments

| | |
|---|---|
| `catField` | vector with categories |
| `ordField` | vector with the scores |
| `categories` | vector, optional. the categories to use from catField |
| `levels` | vector, optional. the levels or order used in ordField. |
| `test` | string, optional. the test of independence to use. Default is "pearson". Other options are "pearson", "fisher", "freeman-tukey", "g", "mod-log", "neyman", "power" |
| `cc` | : string, optional. method for continuity correction. Either NULL (default), "yates", "pearson", "williams" |
| `lambd` | float or string, optional. either name of test or specific value. Default is "cressie-read" i.e. lambda of 2/3. Only applies to Power Divergence test. Other options include float, "cressie-read", "likelihood-ratio", "mod-log", "pearson", "freeman-tukey", "neyman" |

## Details

The Mood Median test creates a 2xk cross table, with k being the number of categories. The two rows are one for the number of scores in that category that are above the overall median, and the second row the number of scores in that category that are equal or below the overall median.

A chi-square test of independence on this cross table can then be performed. There are quite some different options for this:

- "pearson", will perform a Pearson chi-square test of independence using the ts_pearson_ind() function.
- "fisher", will perform a Fisher exact test using the ts_fisher() function, but only if there are 2 categories, if there are more the test will be set to "pearson"
- "freeman-tukey", will perform a Freeman-Tukey test of independence using the ts_freeman_tukey_ind() function

- "g", will perform a G test of independence using the ts_g_ind() function
- "mod-log", will perform a Mod-Log Likelihood test of independence using the ts_mod_log_likelihood_ind() function
- "neyman", will perform a Neyman test of independence using the ts_neyman_ind() function
- "power", will perform a Power Divergence test of independence using the ts_powerdivergence_ind() function.

The formula using the default Pearson test is:

$$\chi_M^2 = \sum_{i=1}^{2}\sum_{j=1}^{k}\frac{\left(F_{i,j} - E_{i,j}\right)^2}{E_{i,j}}$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_M^2, df\right)$$

With:

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{k} F_{i,j}$$

$$C_j = \sum_{i=1}^{2} F_{i,j}$$

$$n = \sum_{i=1}^{2}\sum_{j=1}^{k} F_{i,j} = \sum_{i=1}^{2} R_i = \sum_{j=1}^{k} C_j$$

The original source for the formula is most likely Mood (1950), but the ones shown are based on Brown and Mood (1951).

*Symbols used:*

- $k$, the number of categories (columns)
- $F_{1,j}$, the number of scores is category j that are above the overall median
- $F_{2,j}$, the number of scores is category j that are equal or below the overall median
- $E_{i,j}$, the expected count in row i and column j.
- $R_i$, the row total of row i
- $C_j$, the column total of column j
- $n$, the overall total.
- $df$, the degrees of freedom
- $\chi^2\left(\ldots\right)$, the cumulative distribution function of the chi-square distribution.

## Value

A dataframe with the results of the specified test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Brown, G. W., & Mood, A. M. (1951). On median tests for linear hypotheses. Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, 2, 159–167.

Mood, A. M. (1950). *Introduction to the theory of statistics*. McGraw-Hill.

---

# `ts_multinomial_gof` *Exact Multinomial Test of Goodness-of-Fit*

## Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is a Pearson chi-square test, but also a G-test, Freeman-Tukey, Neyman, Mod-Log Likelihood and Cressie-Read test are possible.

McDonald (2014, p. 82) suggests to always use this exact test as long as the sample size is less than 1000 (which was just picked as a nice round number, when n is very large the exact test becomes computational heavy even for computers).

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_multinomial_gof(data, expCounts = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |

## Details

The exact multinomial test of goodness of fit is done in four steps

Step 1: Determine the probability of the observed counts using the probability mass function of the multinomial distribution

Step 2: Determine all possible permutations with repetition that create a sum equal to the sample size over the k-categories.

Step 3: Determine the probability of each of these permutations using the probability mass function of the multinomial distribution.

Step 4: Sum all probabilities found in step 3 that are equal or less than the one found in step 1.

## Value

Dataframe with:

| | |
|---|---|
| `pObs` | probability of the observed data |
| `ncomb` | number of combinations used |
| `pValue` | two-sided p-value |
| `test used` | a description of the test used |

### Before, After and Alternatives

BBefore this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you want to perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1[1:20, 'mar1']
ts_multinomial_gof(ex1)

#Example 2: dataframe with various settings
ex2 = df1[1:20, 'mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_multinomial_gof(ex2, expCounts=eCounts)

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_multinomial_gof(ex3)
```

---

## `ts_neyman_gof` *Neyman Test of Goodness-of-Fit*

### Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test, but also an exact multinomial, G-test, Freeman-Tukey, Mod-Log Likelihood, Cressie-Read, and Freeman-Tukey-Read test are possible.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_neyman_gof(
  data,
  expCounts = NULL,
  cc = c("none", "yates", "pearson", "williams")
)
```

## Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "yates2", "pearson", or "williams" |

## Details

The formula used is (Neyman, 1949, p. 250):

$$\chi_N^2 = \sum_{i=1}^{k}\frac{\left(O_i - E_i\right)^2}{O_i}$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_N^2, df\right)$$

With:

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

If no expected counts provided:

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

else:

$$E_i = n \times \frac{E_{p_i}}{n_p}$$

$$n_p = \sum_{i=1}^{k} E_{p_i}$$

*Symbols used:*

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies

- $n_p$ the sum of all provided expected counts
- $\chi^2(\dots)$ the chi-square cumulative density function

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

$$\chi^2_{NY} = \sum_{i=1}^{k} \frac{\left(|F_i - E_i| - 0.5\right)^2}{O_i}$$

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$\chi^2_{NY} = \sum_{i=1}^{k} \frac{\max\left(0, \left(|F_i - E_i| - 0.5\right)\right)^2}{O_i}$$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi^2_{NP} = \chi^2_N \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi^2_{NW} = \frac{\chi^2_N}{q}$$

With:

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

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `testUsed` | a description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

Neyman, J. (1949). Contribution to the theory of the chi-square test. *Berkeley Symposium on Math. Stat, and Prob*, 239–273. doi:10.1525/9780520327016-030

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. doi:10.2307/2332518

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
ts_neyman_gof(ex1)

#Example 2: dataframe with various settings
ex2 = df1['mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_neyman_gof(ex2, expCounts=eCounts, cc="yates")
ts_neyman_gof(ex2, expCounts=eCounts, cc="pearson")
ts_neyman_gof(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_neyman_gof(ex3)
```

---

# `ts_neyman_ind` *Neyman Test of Independence*

## Description

This test is similar as a Pearson Chi-Square test of independence. If the significance of this test is below 0.05 (or another pre-defined threshold), the two nominal variables have a significant association.

The test compares the observed counts of the cross table with the so-called expected counts. The expected values are the number of respondents you would expect if the two variables would be independent. See the Pearson Chi-Square test of independence for more details on expected counts.

One problem though is that the test should only be used if not too many cells have a so-called expected count, of less than 5, and the minimum expected count is at least 1. So you will also have to check first if these conditions are met. Most often 'not too many cells' is fixed at no more than 20% of the cells. This is often referred to as 'Cochran conditions', after Cochran (1954, p. 420). Note that for example Fisher (1925, p. 83) is more strict, and finds that all cells should have an expected count of at least 5.

## Usage

```
ts_neyman_ind(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  cc = NULL
)
```

## Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `cc` | optional methdod for continuity correction. Either NULL (default), "yates", "pearson", "williams". |

## Details

The formula used is (Neyman, 1949, p. 250):

$$\chi_N^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{\left(F_{i,j} - E_{i,j}\right)^2}{F_{i,j}}$$

$$df = (r-1)\times(c-1)$$

$$sig. = 1 - \chi^2\left(\chi_N^2, df\right)$$

With:

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

*Symbols used:*

- $r$ the number of categories in the first variable (the number of rows)
- $c$ the number of categories in the second variable (the number of columns)
- $F_{i,j}$ the observed count in row i and column j
- $E_{i,j}$ the expected count in row i and column j
- $R_i$ the i-th row total
- $C_j$ the j-th column total
- $n$ the sum of all counts

- $\chi^2\left(\ldots\right)$ the chi-square cumulative density function

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

Use instead of $F_{i,j}$ the adjusted version defined by:

$$F_{i,j}^* = \begin{cases} F_{i,j} - 0.5 & \text{if } F_{i,j} > E_{i,j} \\ F_{i,j} & \text{if } F_{i,j} = E_{i,j} \\ F_{i,j} + 0.5 & \text{if } F_{i,j} < E_{i,j} \end{cases}$$

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_P^2 \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{PW}^2 = \frac{\chi_P^2}{q}$$

With:

$$q = 1 + \frac{\left(n \times \left(\sum_{i=1}^r \frac{1}{R_i}\right) - 1\right) \times \left(n \times \left(\sum_{j=1}^c \frac{1}{C_j}\right) - 1\right)}{6 \times n \times \mathit{df}}$$

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `n rows` | number of categories used in first field |
| `n col.` | number of categories used in second field |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |
| `min. exp.` | the minimum expected count |
| `prop. exp. below 5` | proportion of cells with expected count less than 5 |
| `test` | description of the test used |

## References

Cochran, W. G. (1954). Some methods for strengthening the common $\chi^2$ tests. *Biometrics, 10*(4), 417. doi:10.2307/3001616

Fisher, R. A. (1925). *Statistical methods for research workers*. Oliver and Boyd.

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

Neyman, J. (1949). Contribution to the theory of the chi-square test. Berkeley Symposium on Math. Stat, and Prob, 239–273. doi:10.1525/9780520327016-030

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

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## `ts_ozdemir_kurt_owa` *Özdemir-Kurt Test*

### Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

### Usage

```
ts_ozdemir_kurt_owa(nomField, scaleField, categories = NULL)
```

### Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

### Details

The formula used (Özdemir & Kurt, 2006, pp. 85-86):

$$B^2 = \sum_{j=1}^{k}\left(c_j \times \sqrt{\ln\left(1 + \frac{t_j^2}{v_i}\right)}\right)^2$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(B^2, df\right)$$

With:

$$\chi_{crit}^2 = Q\left(chi_{crit}^2\left(1 - \alpha, df\right)\right)$$

$$t_j = \frac{\bar{x}_j - \bar{x}_w}{\sqrt{\frac{s_j^2}{n_j}}}$$

$$c_j = \frac{4 \times v_j^2 + \frac{5 \times \left(2 \times z_{crit}^2 + 3\right)}{24}}{4 \times v_j^2 + v_j + \frac{4 \times z_{crit}^2 + 9}{12}} \times \sqrt{v_j}$$

$$v_j = n_j - 1$$

$$\bar{x}_w = \sum_{j=1}^{k} h_j \times \bar{x}_j$$

$$h_j = \frac{w_j}{w}$$

$$w_j = \frac{n_j}{s_j^2}$$

$$w = \sum_{j=1}^{k} w_j$$

$$z_{crit} = Q\left(\Phi\left(1 - \frac{\alpha}{2}\right)\right)$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{j=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols used:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $w_j^*$ the modified weight for category j
- $h_j^*$ the adjusted modified weight for category j
- $df$ the degrees of freedom
- $\alpha$ the significance level (usually 0.05)
- $Q\left(\dots\right)$ the quantile (inverse) distribution function
- $\Phi\left(\dots\right)$ the cumulative density function of the standard normal distribution
- $\chi^2\left(\dots\right)$ the cumulative density function of the chi-square distribution

A binary search for a p-value is done such that $B^2 = \chi^2_{crit}$.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the test statistic (B2 value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Özdemir, A. F., & Kurt, S. (2006). One way fixed effect analysis of variance under variance heterogeneity and a solution proposal. *Selçuk Journal of Applied Mathematics, 7*(2), 81–90.

---

## `ts_pearson_gof` *Pearson Chi-Square Test of Goodness-of-Fit*

---

### Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis). If the test has a p-value below a pre-defined threshold (usually 0.05) the assumption they are all equal in the population will be rejected.

There are quite a few tests that can do this. Perhaps the most commonly used is this Pearson chi-square test, but also an exact multinomial, G-test, Freeman-Tukey, Neyman, Mod-Log Likelihood and Cressie-Read test are possible.

The test compares the observed counts with the expected counts. It is often recommended not to use it if the expected count is at least 5 (Peck & Devore, 2012, p. 593).

A YouTube video with explanation on this test is available here

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_pearson_gof(
  data,
  expCounts = NULL,
  cc = c("none", "yates", "yates2", "pearson", "williams")
)
```

### Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "yates2", "pearson", or "williams" |

### Details

The formula used is (Pearson, 1900):

$$\chi_P^2 = \sum_{i=1}^{k}\frac{\left(O_i - E_i\right)^2}{E_i}$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_P^2, df\right)$$

With:

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

If no expected counts provided:

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

else:

$$E_i = n \times \frac{E_{p_i}}{n_p}$$

$$n_p = \sum_{i=1}^{k} E_{p_i}$$

*Symbols used:*

- $k$ the number of categories
- $F_i$ the (absolute) frequency of category i
- $E_i$ the expected frequency of category i
- $E_{p_i}$ the provided expected frequency of category i
- $n$ the sample size, i.e. the sum of all frequencies
- $n_p$ the sum of all provided expected counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

$$\chi_{PY}^2 = \sum_{i=1}^{k} \frac{\left(\left|F_i - E_i\right| - 0.5\right)^2}{E_i}$$

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$\chi_{PY}^2 = \sum_{i=1}^{k} \frac{\max\left(0, \left(\left|F_i - E_i\right| - 0.5\right)\right)^2}{E_i}$$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_{P}^2 \times \frac{n - 1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi_{PW}^2 = \frac{\chi_{P}^2}{q}$$

With:

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

The formula is also used by McDonald (2014, p. 87)

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `test used` | a description of the test used |

### Before, After and Alternatives

Before this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might an effect size measure: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

or perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Allen, A. O. (1990). *Probability, statistics, and queueing theory with computer science applications* (2nd ed.). Academic Press.

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

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. doi:10.2307/2332518

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine Series 5, 50*(302), 157–175. doi:10.1080/14786440009463897

Peck, R., & Devore, J. L. (2012). *Statistics: The exploration and analysis of data* (7th ed). Brooks/Cole.

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
#Example 1: dataframe
ex1 = df1['mar1']
ts_pearson_gof(ex1)

#Example 2: dataframe with various settings
ex2 = df1['mar1']
```

```
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_pearson_gof(ex2, expCounts=eCounts, cc="yates")
ts_pearson_gof(ex2, expCounts=eCounts, cc="pearson")
ts_pearson_gof(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_pearson_gof(ex3)
```

---

## `ts_pearson_ind` *Pearson Chi-Square Test of Independence*

### Description

To test if two nominal variables have an association, the most commonly used test is the Pearson chi-square test of independence (Pearson, 1900). If the significance of this test is below 0.05, the two nominal variables have a significant association.

The test compares the observed counts of the cross table with the so-called expected counts. The expected values are the number of respondents you would expect if the two variables would be independent.

If for example I had 50 male and 50 female respondents, and 50 agreed with a statement and 50 disagreed with the statement, the expected value for each combination (male-agree, female-agree, male-disagree, and female-disagree) would be 25.

Note that if in the survey the real results would be that all male disagreed, and all female would agree, there is a full dependency (i.e. gender fully decides if you agree or disagree), even though the row and column totals would still be 50. In essence the Pearson chi-square test, checks if your data is more toward the expected values (independence) or the full dependency one.

One problem though is that the Pearson chi-square test should only be used if not too many cells have a so-called expected count, of less than 5, and the minimum expected count is at least 1. So you will also have to check first if these conditions are met. Most often ‘not too many cells’ is fixed at no more than 20% of the cells. This is often referred to as ’Cochran conditions’, after Cochran (1954, p. 420). Note that for example Fisher (1925, p. 83) is more strict, and finds that all cells should have an expected count of at least 5 .

### Usage

```
ts_pearson_ind(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  cc = NULL
)
```

## Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `cc` | optional methdod for continuity correction. Either NULL (default), "yates", "pearson", "williams". |

## Details

The formula used is (Pearson, 1900, p. 165):

$$\chi_p^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{\left(F_{i,j} - E_{i,j}\right)^2}{E_{i,j}}$$

$$df = (r - 1) \times (c - 1)$$

$$sig. = 1 - \chi^2\left(\chi_p^2, df\right)$$

With:

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j} = \sum_{i=1}^{r} R_i = \sum_{j=1}^{c} C_j$$

Symbols:

- $r$, the number of rows
- $c$, the number of columns
- $F_{i,j}$, the observed count in row i and column j.
- $E_{i,j}$, the expected count in row i and column j.
- $R_i$, the row total of row i
- $C_j$, the column total of column j
- $n$, the overall total.
- $df$, the degrees of freedom

The **Yates** correction uses $F'_{i,j}$ instead of $F_{i,j}$, defined as (Yates, 1934, p. 222):

$$F'_{i,j} = \begin{cases} F_{i,j} - \frac{1}{2} & \text{if } F_{i,j} > E_{i,j} \\ F_{i,j} + \frac{1}{2} & \text{if } F_{i,j} < E_{i,j} \\ F_{i,j} & \text{if } F_{i,j} = E_{i,j} \end{cases}$$

The **Williams** correction, adjusts the Pearson chi-square value:

$$\chi^2_{wil} = \frac{\chi^2_p}{q}$$

With:

$$q = 1 + \frac{\left(n\times\left(\sum_{i=1}^r \frac{1}{R_i}\right)-1\right)\times\left(n\times\left(\sum_{j=1}^c \frac{1}{C_i}\right)-1\right)}{6\times n\times(r-1)\times(c-1)}$$

The formula is probably from Williams (1976) but the one shown here is taken from McDonald (1976, p. 36).

The **Pearson** correction also adjusts the Pearson chi-square value with (E.S. Pearson, 1947, p. 157):

$$\chi^2_{epearson} = \frac{n-1}{n}\times\chi^2_p$$

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `n rows` | number of categories used in first field |
| `n col.` | number of categories used in second field |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |
| `min. exp.` | the minimum expected count |
| `prop. exp. below 5` | proportion of cells with expected count less than 5 |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cochran, W. G. (1954). Some methods for strengthening the common $\chi^2$ tests. *Biometrics, 10*(4), 417. doi:10.2307/3001616

Fisher, R. A. (1925). *Statistical methods for research workers*. Oliver and Boyd.

McDonald, J. H. (2014). *Handbook of biological statistics* (3rd ed.). Sparky House Publishing.

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. doi:10.2307/2332518

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine Series 5, 50*(302), 157–175. doi:10.1080/14786440009463897

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

---

`ts_powerdivergence_gof`

## *Power Divergence Goodness-of Fit Tests*

---

### Description

A test that can be used with a single nominal variable, to test if the probabilities in all the categories are equal (the null hypothesis)

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test ($\chi^2$), but also an exact multinomial, G-test ($G^2$), Freeman-Tukey ($T^2$), Neyman ($NM^2$), Mod-Log Likelihood ($GM^2$), and Freeman-Tukey-Read test are possible.

Cressie and Read (1984, p. 463) noticed how the $\chi^2$, $G^2$, $T^2$, $NM^2$ and $GM^2$ can all be captured with one general formula. The additional variable lambda ($\lambda$) was then investigated, and they settled on a $\lambda$ of 2/3.

By setting $\lambda$ to different values, we get the different tests:

- $\lambda = 1$Pearson chi-square
- $\lambda = 0$G/Wilks/Likelihood-Ratio
- $\lambda = -\frac{1}{2}$Freeman-Tukey
- $\lambda = -1$Mod-Log-Likelihood
- $\lambda = -2$Neyman
- $\lambda = \frac{2}{3}$Cressie-Read

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_powerdivergence_gof(
  data,
  expCounts = NULL,
  lambd = c("cressie-read", "g", "mod-log", "freeman-tukey", "neyman"),
  cc = c("none", "yates", "yates2", "pearson", "williams")
)
```

### Arguments

| | |
|---|---|
| `data` | A vector or dataframe with the data |
| `expCounts` | Optional dataframe with the categories and expected counts |
| `lambd` | Optional either name of test or specific value. Either "cressie-read" (default), "g", "mod-log", "freeman-tukey", or "neyman" |
| `cc` | Optional continuity correction. Either "none" (default), "yates", "pearson", or "williams" |

## Details

The formula used is (Cressie & Read, 1984, p. 442):

$$\chi_C^2 = \begin{cases} 2 \times \sum_{i=1}^{k} F_i \times ln\left(\frac{F_i}{E_i}\right) & \text{if } \lambda = 0 \\ 2 \times \sum_{i=1}^{k} E_i \times ln\left(\frac{E_i}{F_i}\right) & \text{if } \lambda = -1 \\ \frac{2}{\lambda \times (\lambda + 1)} \times \sum_{i=1}^{k} F_i \times \left(\left(\frac{F_i}{E_i}\right)^{\lambda} - 1\right) & \text{else} \end{cases}$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(\chi_C^2, df\right)$$

With:

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$

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

*Symbols used:*

- $k$ the number of categories
- $F_i$ the observed count of category i
- $E_i$ the expected count of category i
- $n$ the sum of all counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

Cressie and Read (1984, p. 463) suggest to use $\lambda = \frac{2}{3}$, which is therefor the default in this function.

The **Pearson chi-square statistic** can be obtained by setting $\lambda = 1$.

The **Freeman-Tukey test** will be same as setting lambda to $-\frac{1}{2}$.

**Neyman test** will be same as setting lambda to $-2$.

The Yates continuity correction (cc="yates") is calculated using (Yates, 1934, p. 222):

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

In some cases the Yates correction is slightly changed to (yates2) (Allen, 1990, p. 523):

$$F_i^* = \begin{cases} F_i - 0.5 & \text{if } F_i - 0.5 > E_i \\ F_i + 0.5 & \text{if } F_i + 0.5 < E_i \\ F_i & \text{else} \end{cases}$$

Note that the Yates correction is usually only considered if there are only two categories. Some also argue this correction is too conservative (see for details Haviland (1990)).

The Pearson correction (cc="pearson") is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{adj}^2 = \chi_C^2 \times \frac{n-1}{n}$$

The Williams correction (cc="williams") is calculated using (Williams, 1976, p. 36):

$$\chi^2_{adj} = \frac{\chi^2_C}{q}$$

With:

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

The formula is also used by McDonald (2014, p. 87)

## Value

Dataframe with:

| | |
|---|---|
| `statistic` | the chi-square statistic |
| `df` | the degrees of freedom |
| `pValue` | two-sided p-value |
| `minExp` | the minimum expected count |
| `percBelow5` | the percentage of expected counts below 5 |
| `test used` | a description of the test used |

## Before, After and Alternatives

BBefore this an impression using a frequency table or a visualisation might be helpful: `tab_frequency`, for a frequency table `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

After this you might an effect size measure: `es_cohen_w`, for Cohen w. `es_cramer_v_gof`, for Cramer's V for Goodness-of-Fit. `es_fei`, for Fei. `es_jbm_e`, for Johnston-Berry-Mielke E.

or perform a post-hoc test: `ph_pairwise_bin`, for Pairwise Binary Tests. `ph_pairwise_gof`, for Pairwise Goodness-of-Fit Tests. `ph_residual_gof_bin`, for Residuals Tests using Binary tests. `ph_residual_gof_gof`, for Residuals Using Goodness-of-Fit Tests.

Alternative tests: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Bishop, Y. M. M., Fienberg, S. E., & Holland, P. W. (2007). *Discrete multivariate analysis*. Springer.

Cressie, N., & Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society: Series B (Methodological), 46*(3), 440–464. doi:10.1111/j.2517-6161.1984.tb01318.x

Freeman, M. F., & Tukey, J. W. (1950). Transformations related to the angular and the square root. *The Annals of Mathematical Statistics, 21*(4), 607–611. doi:10.1214/aoms/1177729756

Haviland, M. G. (1990). Yates's correction for continuity and the analysis of 2 × 2 contingency tables. *Statistics in Medicine, 9*(4), 363–367. doi:10.1002/sim.4780090403

Neyman, J. (1949). Contribution to the theory of the chi-square test. Berkeley Symposium on Math. Stat, and Prob, 239–273. doi:10.1525/9780520327016-030

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. doi:10.2307/2332518

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine Series 5, 50*(302), 157–175. doi:10.1080/14786440009463897

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. doi:10.1214/aoms/1177732360

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
ts_powerdivergence_gof(ex1)

#Example 2: dataframe with various settings
ex2 = df1['mar1']
eCounts = data.frame(c("MARRIED", "DIVORCED", "NEVER MARRIED", "SEPARATED"), c(5,5,5,5))
ts_powerdivergence_gof(ex2, expCounts=eCounts)
ts_powerdivergence_gof(ex2, expCounts=eCounts, cc="pearson")
ts_powerdivergence_gof(ex2, expCounts=eCounts, cc="williams")

#Example 3: a list
ex3 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
ts_powerdivergence_gof(ex3)
```

---

`ts_powerdivergence_ind`

# *Power Divergence Test of Independence*

## Description

A test that can be used with two nominal variables to test if they are independent.

There are quite a few tests that can do this. Perhaps the most commonly used is the Pearson chi-square test ($\chi^2$), but also an exact multinomial, G-test ($G^2$), Freeman-Tukey ($T^2$), Neyman ($NM^2$), Mod-Log Likelihood ($GM^2$), and Freeman-Tukey-Read test are possible.

Cressie and Read (1984, p. 463) noticed how the $\chi^2$, $G^2$, $T^2$, $NM^2$ and $GM^2$ can all be captured with one general formula. The additional variable lambda ($\lambda$) was then investigated, and they settled on a $\lambda$ of 2/3.

By setting $\lambda$ to different values, we get the different tests:

- $\lambda = 1$Pearson chi-square
- $\lambda = 0$G/Wilks/Likelihood-Ratio
- $\lambda = -\frac{1}{2}$Freeman-Tukey
- $\lambda = -1$Mod-Log-Likelihood
- $\lambda = -2$Neyman
- $\lambda = \frac{2}{3}$Cressie-Read

## Usage

```
ts_powerdivergence_ind(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  cc = NULL,
  lambd = 2/3
)
```

## Arguments

| | |
|---|---|
| `field1` | list or dataframe with the first categorical field |
| `field2` | list or dataframe with the second categorical field |
| `categories1` | optional list with order and/or selection for categories of field1 |
| `categories2` | optional list with order and/or selection for categories of field2 |
| `cc` | optional methdod for continuity correction. Either NULL (default), "yates", "pearson", "williams". |
| `lambd` | Optional either name of test or specific value. Default is "cressie-read" i.e. lambda of 2/3 |

## Details

The formula used is (Cressie & Read, 1984, p. 442):

$$\chi_C^2 = \begin{cases} 2 \times \sum_{i=1}^{r} \sum_{j=1}^{c} \left(F_{i,j} \times ln\left(\frac{F_{i,j}}{E_{i,j}}\right)\right) & \text{if } \lambda = 0 \\ 2 \times \sum_{i=1}^{r} \sum_{j=1}^{c} \left(E_{i,j} \times ln\left(\frac{E_{i,j}}{F_{i,j}}\right)\right) & \text{if } \lambda = -1 \\ \frac{2}{\lambda\times(\lambda+1)} \times \sum_{i=1}^{r} \sum_{j=1}^{c} F_{i,j} \times \left(\left(\frac{F_{i,j}}{E_{i,j}}\right)^{\lambda} - 1\right) & \text{else} \end{cases}$$

$$df = (r-1) \times (c-1)$$

$$sig. = 1 - \chi^2\left(\chi_C^2, df\right)$$

With:

$$n = \sum_{i=1}^{r} \sum_{j=1}^{c} F_{i,j}$$

$$E_{i,j} = \frac{R_i \times C_j}{n}$$

$$R_i = \sum_{j=1}^{c} F_{i,j}$$

$$C_j = \sum_{i=1}^{r} F_{i,j}$$

*Symbols used:*

- $r$ the number of categories in the first variable (the number of rows)
- $c$ the number of categories in the second variable (the number of columns)
- $F_{i,j}$ the observed count in row i and column j
- $E_{i,j}$ the expected count in row i and column j
- $R_i$ the i-th row total
- $C_j$ the j-th column total
- $n$ the sum of all counts
- $\chi^2\left(\dots\right)$ the chi-square cumulative density function

Cressie and Read (1984, p. 463) suggest to use $\lambda = \frac{2}{3}$, which is therefor the default in this function.

The **Pearson chi-square statistic** can be obtained by setting $\lambda = 1$. Pearson's original formula is (Pearson, 1900, p. 165):

$$\chi_P^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{\left(F_{i,j} - E_{i,j}\right)^2}{E_{i,j}}$$

The **Freeman-Tukey test** has as a formula (Bishop et al., 2007, p. 513):

$$T^2 = 4 \times \sum_{i=1}^{r}\sum_{j=1}^{c}\left(\sqrt{F_{i,j}} - \sqrt{E_{i,j}}\right)^2$$

This will be same as setting lambda to $-\frac{1}{2}$. Note that the source for the formula is often quoted to be from Freeman and Tukey (1950) but couldn't really find it in that article.

**Neyman test** formula was very similar to Pearson's, but the observed and expected counts swapped (Neyman, 1949, p. 250):

$$\chi_N^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{\left(E_{i,j} - F_{i,j}\right)^2}{F_{i,j}}$$

This will be same as setting lambda to $-2$.

The Yates correction (yates) is calculated using (Yates, 1934, p. 222):

Use instead of $F_{i,j}$ the adjusted version defined by:

$$F_{i,j}^* = \begin{cases} F_{i,j} - 0.5 & \text{if } F_{i,j} > E_{i,j} \\ F_{i,j} & \text{if } F_{i,j} = E_{i,j} \\ F_{i,j} + 0.5 & \text{if } F_{i,j} < E_{i,j} \end{cases}$$

The Pearson correction (pearson) is calculated using (E.S. Pearson, 1947, p. 157):

$$\chi_{PP}^2 = \chi_P^2 \times \frac{n-1}{n}$$

The Williams correction (williams) is calculated using (Williams, 1976, p. 36):

$$\chi^2_{PW} = \frac{\chi^2_P}{q}$$

With:

$$q = 1 + \frac{\left(n \times \left(\sum_{i=1}^{r} \frac{1}{R_i}\right) - 1\right) \times \left(n \times \left(\sum_{j=1}^{c} \frac{1}{C_j}\right) - 1\right)}{6 \times n \times df}$$

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `n rows` | number of categories used in first field |
| `n col.` | number of categories used in second field |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |
| `min. exp.` | the minimum expected count |
| `prop. exp. below 5` | proportion of cells with expected count less than 5 |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel

## References

Bishop, Y. M. M., Fienberg, S. E., & Holland, P. W. (2007). *Discrete multivariate analysis*. Springer.

Cressie, N., & Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society: Series B (Methodological), 46*(3), 440–464. https://doi.org/10.1111/j.2517-6161.1984.tb01318.x

Freeman, M. F., & Tukey, J. W. (1950). Transformations related to the angular and the square root. *The Annals of Mathematical Statistics, 21*(4), 607–611. https://doi.org/10.1214/aoms/1177729756

Neyman, J. (1949). Contribution to the theory of the chi-square test. Berkeley Symposium on Math. Stat, and Prob, 239–273. https://doi.org/10.1525/9780520327016-030

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

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine Series 5, 50*(302), 157–175. https://doi.org/10.1080/14786440009463897

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

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. https://doi.org/10.2307/2983604

---

## `ts_score_os` *One-Sample Score Test*

---

### Description

A one-sample score test could be used with binary data, to test if the two categories have a significantly different proportion. It is an approximation of a binomial test, by using a standard normal distribution. Since the binomial distribution is discrete while the normal is continuous, a so-called continuity correction can (should?) be applied.

The null hypothesis is usually that the proportions of the two categories in the population are equal (i.e. 0.5 for each). If the p-value of the test is below the pre-defined alpha level (usually 5% = 0.05) the null hypothesis is rejected and the two categories differ in proportion significantly.

The input for the function doesn't have to be a binary variable. A nominal variable can also be used and the two categories to compare indicated.

A significance in general is the probability of a result as in the sample, or more extreme, if the null hypothesis is true.

Some info on the different tests can be found in video. This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_score_os(
  data,
  p0 = 0.5,
  p0Cat = NULL,
  codes = NULL,
  cc = c("none", "yates")
)
```

### Arguments

| | |
|---|---|
| `data` | A vector or dataframe with the data |
| `p0` | Optional hypothesized proportion for the first category (default is 0.5) |
| `p0Cat` | Optional the category for which p0 was used |
| `codes` | Optional vector with the two codes to use |
| `cc` | optional use of continuity correction. Either "none" (default) or "Yates". |

### Details

Also sometimes called a 'proportion' test.

To decide on which category is associated with p0 the following is used:

- If codes are provided, the first code is assumed to be the category for the p0.
- If p0Cat is specified that will be used for p0 and all other categories will be considered as category 2, this means if there are more than two categories the remaining two or more (besides p0Cat) will be merged as one large category.
- If neither codes or p0Cat is specified and more than two categories are in the data a warning is printed and no results.

- If neither codes or p0Cat is specified and there are two categories, p0 is assumed to be for the category closest matching the p0 value (i.e. if p0 is above 0.5 the category with the highest count is assumed to be used for p0)

The formula used is (Wilson, 1927):

$$z = \frac{x - \mu}{SE}$$

With:

$$\mu = n \times p_0$$

$$SE = \sqrt{\mu \times (1 - p_0)}$$

*Symbols used:*

- $x$ is the number of successes in the sample
- $p_0$ the expected proportion (i.e. the proportion according to the null hypothesis)

If the Yates continuity correction is used the formula changes to (Yates, 1934, p. 222):

$$z_{Yates} = \frac{|x - \mu| - 0.5}{SE}$$

The formula used and naming comes from IBM (2021, p. 997) who refer to Agresti, most likeli Agresti (2013, p. 10)

### Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the test value |
| `pValue` | two-sided p-value |
| `test` | a description of the test used |

### Before, After and Alternatives

Before running the test you might first want to get an impression using a frequency table: tab_frequency

After the test you might want an effect size measure: `es_cohen_g`, for Cohen g `es_cohen_h_os`, for Cohen h' `es_alt_ratio`, for Alternative Ratio

Alternatives for this test could be: `ts_binomial_os`, for One-Sample Binomial Test `ts_wald_os`, for One-Sample Wald Test

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Agresti, A. (2013). *Categorical data analysis* (3rd ed.). Wiley.

IBM SPSS Statistics Algorithms. (2021). IBM.

Wilson, E. B. (1927). Probable Inference, the Law of Succession, and Statistical Inference. *Journal of the American Statistical Association, 22*(158), 209–212. doi:10.2307/2276774

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## Examples

```
#Example 1: Numeric list
ex1 = c(1, 1, 2, 1, 2, 1, 2, 1)
ts_score_os(ex1)
ts_score_os(ex1, p0=0.3)
ts_score_os(ex1, p0=0.3, cc="yates")

#Example 2: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ts_score_os(df1['sex'])
ts_score_os(df1['mar1'], codes=c("DIVORCED", "NEVER MARRIED"))
```

---

# `ts_scott_smith_owa` *Scott-Smith Test*

---

## Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

Yiğit and Gökpina (2010, p. 32) concluded that this test is inferior to some other alternatives when there is heteroscedasticity (variances in the groups not the same) are preferred (for example the Welch one-way ANOVA).

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

## Usage

```
ts_scott_smith_owa(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used (Scott & Smith, 1971, p. 277):

$$\chi^2_{SS} = \sum_{j=1}^{k} d_j^2$$

$$df = k$$

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

With:

$$d_j = t_j \times \sqrt{\frac{n_j - 3}{n_j - 1}}$$

$$t_j = \frac{\left(\bar{x}_j - \bar{x}\right) \times \sqrt{n_j}}{s_j}$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j} \left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

$$\bar{x} = \frac{\sum_{j=1}^{k} n_j \times \bar{x}_j}{n}$$

$$n = \sum_{j=1}^{k} n_j$$

*Symbols:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $x_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $df$ the degrees of freedom
- $\chi^2\left(\dots, \dots\right)$ the cumulative distribution function of the chi-square distribution.

I couldn't find the chi-square test itself in the original article, but the calculation for an independent samples test repeated usually indeed leads to a chi-square distribution. The formula and chi-square distribution can also be found in Adepoju et al. (2016, p. 64), Cavus and Yazici (2020, p. 7) and Yiğit and Gökpina (2010, p. 17)

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `k` | the number of categories |
| `statistic` | the test statistic (chi-square value) |
| `df` | the degrees of freedom |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Adepoju, K. A., Shittu, O. I., & Chukwu, A. U. (2016). On the development of an exponentiated F test for one-way ANOVA in the presence of outlier(s). *Mathematics and Statistics, 4*(2), 62–69. https://doi.org/10.13189/ms.2016.040203

Cavus, M., & Yazıcı, B. (2020). Testing the equality of normal distributed and independent groups' means under unequal variances by doex package. *The R Journal, 12*(2), 134. https://doi.org/10.32614/RJ-2021-008

Scott, A. J., & Smith, T. M. F. (1971). Interval estimates for linear combinations of means. *Applied Statistics, 20*(3), 276–285. https://doi.org/10.2307/2346757

Yiğit, E., & Gökpinar, F. (2010). A simulation study on tests for one-way ANOVA under the unequal variance assumption. *Communications, Faculty Of Science, University of Ankara*, 15–34. https://doi.org/10.1501/Commua1_0000000660

---

## `ts_sign_os` *one-sample sign test*

### Description

This function will perform one-sample sign test.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_sign_os(data, levels = NULL, mu = NULL)
```

### Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `levels` | optional vector with levels in order |
| `mu` | optional hypothesized median, otherwise the midrange will be used |

### Details

The test statistic is calculated using (Stewart, 1941, p. 236):

$$p = 2 \times B\left(n, \min\left(n_+, n_-\right), \frac{1}{2}\right)$$

*Symbols used:*

- $B\left(\dots\right)$ is the binomial cumulative distribution function
- $n$ is the number of cases
- $n_+$ is the number of cases above the hypothesized median
- $n_-$ is the number of cases below the hypothesized median
- $min$ is the minimum value of the two values

The test is described in Stewart (1941), although there are earlier uses.

The paired version for example was already described by Arbuthnott (1710)

## Value

Dataframe with:

| | |
|---|---|
| `mu` | the mean tested |
| `p-value` | he significance (p-value) |
| `test` | a description of the test used |

## Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want to determine an effect size measure: `es_common_language_os`, for the Common Language Effect Size. `es_dominance`, for the Dominance score. `r_rank_biserial_os`, for the Rank-Biserial Correlation

Alternative tests: `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Arbuthnott, J. (1710). An argument for divine providence, taken from the constant regularity observ'd in the births of both sexes. *Philosophical Transactions of the Royal Society of London, 27*(328), 186–190. https://doi.org/10.1098/rstl.1710.0011

Stewart, W. M. (1941). A note on the power of the sign test. *The Annals of Mathematical Statistics, 12*(2), 236–239. https://doi.org/10.1214/aoms/1177731755

## Examples

```
#Example 1: Text dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
ts_sign_os(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
ts_sign_os(ex2)
```

---

## `ts_sign_ps` *Paired Samples Sign Test*

---

### Description

This test compares the number of pairs that have a difference above the hypothesized difference, with those below the difference. It can be considered an alternative for the paired samples t-test.

### Usage

```
ts_sign_ps(field1, field2, levels = NULL, dmu = 0, method = "exact")
```

### Arguments

| | |
|---|---|
| `field1` | the numeric scores of the first variable |
| `field2` | the numeric scores of the second variable |
| `levels` | vector, optional. the levels from field1 and field2 |
| `dmu` | float, optional. The difference according to the null hypothesis (default is 0) |
| `method` | string, optional. Test to be used. Either "exact" (default), "appr". |

### Details

If method="exact" the binomial distribution will be used. The formula used is (Dixon & Mood, 1946):

$$sig. = 2 \times \text{Bin}\left(n, \min\left(n_{pos}, n_{neg}\right), \frac{1}{2}\right)$$

When using the approximation, the standard normal distribution is used (SPSS, 2006, p. 483):

$$z = \frac{\max\left(n_{pos}, n_{neg}\right) - 0.5 \times \left(n_{pos} + n_{neg}\right) - 0.5}{0.5 \times \sqrt{n_{pos} + n_{neg}}}$$

$$sig. = 2 \times \left(1 - \Phi\left(|z|\right)\right)$$

With:

$$n_{pos} = \sum_{i=1}^{n} \begin{cases} 1 & \text{if } d_i > d_{H0} \\ 0 & \text{if } d_i \leq d_{H0} \end{cases}$$

$$n_{neg} = \sum_{i=1}^{n} \begin{cases} 0 & \text{if } d_i \geq d_{H0} \\ 1 & \text{if } d_i < d_{H0} \end{cases}$$

$$d_i = x_i - y_i$$

*Symbols used:*

- $n$ is the number of pairs with a difference unequal to zero
- $n_{pos}$ the number of pairs with a positive difference
- $n_{neg}$ the number of pairs with a negative difference
- $d_{H0}$ the difference according to the null hypothesis, usually 0
- $x_i$ the i-th score from the first variable

- $y_i$ the i-th score from the second variable
- Bin $(\ldots,\ldots)$ the cumulative probability mass function of the binomial distribution

The test was described by Arbuthnott (1710) but with more modern notation see Dixon and Mood (1946).

**Alternatives**

*library(DescTools)*

SignTest(ord1, ord2)

*library(EnvStats)*

signTest(ord1, ord2, paired=TRUE)

*library(BSDA)*

SIGN.test(ord1, ord2)

## Value

A dataframe with:

| | |
|---|---|
| `n pos` | the number of scores with a positive difference |
| `n neg` | the number of scores with a negative difference |
| `statistic` | the test statistic (only applicable if method="appr") |
| `p-Value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Arbuthnott, J. (1710). An argument for divine providence, taken from the constant regularity observed in the births of both sexes. *Philosophical Transactions of the Royal Society of London, 27*(328), 186–190. doi:10.1098/rstl.1710.0011

Dixon, W. J., & Mood, A. M. (1946). The statistical sign test. *Journal of the American Statistical Association, 41*(236), 557–566. doi:10.1080/01621459.1946.10501898

SPSS. (2006). SPSS 15.0 algorithms.

---

# `ts_stuart_maxwell` *Stuart-Maxwell / Marginal Homogeneity Test*

## Description

If you are only interested if the overall distribution changed (i.e. if the percentages from each category changed or not), you can perform a marginal homogeneity test. There are two that seem to be quite popular for this, the Stuart-Maxwell test (Stuart, 1955; Maxwell, 1970), and the Bhapkar test (Bhapkar, 1961; 1966). According Uebersax (2006) (which also has a nice example) the Bhapkar one is preferred.

Simply put, a marginal homogeneity test, looks at the row vs column proportions. Since in a paired test, the options are the same, if the row and column proportions are the same, nothing changed between the two variables.

## Usage

```
ts_stuart_maxwell(field1, field2, categories = NULL)
```

## Arguments

| | |
|---|---|
| `field1` | vector, the first categorical field |
| `field2` | vector, the first categorical field |
| `categories` | vector, optional, order and/or selection for categories of field1 and field2 |

## Details

The formula used is:

$$\chi_{SM}^2 = n \times d' \times S^{-1} \times d$$

$$df = r - 1 = c - 1$$

$$sig. = 1 - \chi^2\left(\chi_{SM}\right)$$

With:

$$S_{i,i} = p_{i,.} + p_{.,i} - 2 \times p_{i,i}$$

$$S_{i,j} = -\left(p_{i,j} + p_{j,i}\right)$$

$$d_i = p_{i,.} - p_{.,i}$$

$$p_{i,j} = \frac{F_{i,j}}{n}$$

$$d = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_{r-1} \end{bmatrix}$$

$$S = \begin{bmatrix} S_{1,1} & S_{1,2} & \dots & S_{1,c-1} \\ S_{2,1} & S_{2,2} & \dots & S_{2,c-1} \\ \dots & \dots & \dots & \dots \\ S_{r-1,1} & S_{r-1,2} & \dots & S_{r-1,c-1} \end{bmatrix}$$

$$n = \sum_{i=1}^{r}\sum_{j=1}^{c} F_{i,j}$$

*Symbols used*

- $r$ is the number of rows (categories in the first variable)
- $c$ is the number of columns (categories in the second variable)
- $n$ is the total number of scores
- $F_{i,j}$ is the frequency (count) of scores equal to the i-th category in the first variable, and the j-th category in the second.
- $p_{i,.}$ The sum of the proportions in row i
- $p_{.,i}$ The sum of the proportions in column i
- $d'$ is the transpose of the d vector
- $S^{-1}$ is the inverse of the S matrix.
- $\chi^2\left(\dots,\dots\right)$ is the cumulative distribution function of the chi-square distribution

*Note*

- the d vector and S matrix are one row (and column) less.
- This test only differs from the Bhapkar test in the calculation of S
- The procedure is found ins Maxwell (1970) which is founded by Stuart (1955).

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the chi-squared value |
| `df` | the degrees of freedom used in the test |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Bhapkar, V. P. (1961). Some tests for categorical data. *The Annals of Mathematical Statistics, 32*(1), 72–83. doi:10.1214/aoms/1177705140

Bhapkar, V. P. (1966). A note on the equivalence of two test criteria for hypotheses in categorical data. *Journal of the American Statistical Association, 61*(313), 228–235. doi:10.1080/01621459.1966.10502021

Maxwell, A. E. (1970). Comparing the classification of subjects by two independent judges. *The British Journal of Psychiatry, 116*(535), 651–655. doi:10.1192/bjp.116.535.651

Stuart, A. (1955). A test for homogeneity of the marginal distributions in a two-way classification. *Biometrika, 42*(3/4), 412–416. doi:10.2307/2333387

Uebersax, J. (2006, August 30). McNemar tests of marginal homogeneity. http://www.john-uebersax.com/stat/mcnemar.htm

---

# `ts_student_t_is` *Student t Test (Independent Samples)*

## Description

A test to compare two means. The null hypothesis would be that the means of each category are equal in the population.

The test assumes that the variances in the population of the scores are the same. If this is not the case, a Welch t-test could be used. Ruxten (2006) even argues that the Welch t-test should always be prefered over the Student t-test.

There are four similar tests, with different assumptions.

| test | equal variance | normality |
|---|---|---|
| Student | yes | yes |
| Welch | no | yes |
| Trimmed | yes | no |

| | | |
|---|---|---|
| Yuen-Welch | no | no |

The Trimmed and Yuen-Welch can be found in the **ts_trimmed_mean_is()**, and the Welch t-test with the **ts_welch_t_is().**

### Usage

```
ts_student_t_is(catField, scaleField, categories = NULL, dmu = 0)
```

### Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scaleField` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `dmu` | Optional difference according to null hypothesis (default is 0) |

### Details

The formula used is:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{SE}$$

$$df = n_1 + n_2 - 2$$

$$sig. = 2 \times \left(1 - T\left(\left|t\right|, df\right)\right)$$

With:

$$SE = s_p \times \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}$$

$$s_p = \sqrt{\frac{\left(n_1 - 1\right) \times s_1^2 + \left(n_2 - 1\right) \times s_2^2}{df}}$$

$$s_i^2 = \frac{\sum_{j=1}^{n_i} \left(x_{i,j} - \bar{x}_i\right)^2}{n_i - 1}$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{i,j}}{n_i}$$

*Symbols used:*

- $x_{i,j}$ the j-th score in category i
- $n_i$ the number of scores in category i

### Value

A dataframe with:

| | |
|---|---|
| `n cat. 1` | the sample size of the first category |
| `n cat. 2` | the sample size of the second category |
| `mean cat. 1` | the sample mean of the first category |
| `mean cat. 2` | the sample mean of the second category |
| `diff.` | difference between the two sample means |
| `hyp. diff.` | hypothesized difference between the two population means |
| `statistic` | the test statistic (t-value) |
| `df` | degrees of freedom |
| `pValue` | the significance (p-value) |
| `test` | name of test used |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Ruxton, G. D. (2006). The unequal variance t-test is an underused alternative to Student's t-test and the Mann–Whitney U test. *Behavioral Ecology, 17*(4), 688–690. https://doi.org/10.1093/beheco/ark016

Student. (1908). The probable error of a mean. *Biometrika, 6*(1), 1–25. https://doi.org/10.1093/biomet/6.1.1

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['age']
ex1 = replace(ex1, ex1=="89 OR OLDER", "90")
ts_student_t_is(df1['sex'], ex1)

#Example 2: vectors
scores = c(20,50,80,15,40,85,30,45,70,60, NA, 90,25,40,70,65, NA, 70,98,40)
groups = c("nat.","int.","int.","nat.","int.", "int.","nat.","nat.","int.",
"int.","int.","int.","int.","int.","nat.", "int." ,NA,"nat.","int.","int.")
ts_student_t_is(groups, scores)
```

---

## `ts_student_t_os` *One-Sample Student t-Test*

### Description

A test for a single (arithmetic) mean.

The assumption about the population (null hypothesis) for this test is a pre-defined mean, i.e. the (arithmetic) mean that is expected in the population. If the p-value (significance) is then below a pre-defined threhold (usually 0.05), the assumption is rejected.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_student_t_os(data, mu = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `mu` | optional hypothesized mean, otherwise the midrange will be used |

## Details

The formula used is:

$$t = \frac{\bar{x} - \mu_{H_0}}{SE}$$

$$sig = 2 \times \left(1 - T\left(\left|t\right|, df\right)\right)$$

With:

$$df = n - 1$$

$$SE = \frac{s}{\sqrt{n}}$$

$$s = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n - 1}}$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

*Symbols used:*

- $T\left(\dots, \dots\right)$ the cumulative distribution function of the t-distribution
- $\bar{x}$ the sample mean
- $\mu_{H_0}$ the hypothesized mean in the population
- $SE$ the standard error (i.e. the standard deviation of the sampling distribution)
- $df$ the degrees of freedom
- $n$ the sample size (i.e. the number of scores)
- $s$ the unbiased sample standard deviation
- $x_i$ the i-th score

The Student t test (Student, 1908) was described by Gosset under the pseudo name Student.

## Value

A dataframe with:

| | |
|---|---|
| `mu` | the hypothesized mean |
| `sample mean` | sample mean |
| `statistic` | test statistic |
| `df` | degrees of freedom |
| `p-value` | p-value (sig.) |
| `test used` | test used |

### Before, After and Alternatives

Before this you might want to create a binned frequency table or a visualisation: `tab_frequency_bins`, to create a binned frequency table. `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

After this you might want an effect size measure: `es_cohen_d_os`, for for Cohen d'. `es_hedges_g_os`, for Hedges g. `es_common_language_os`, for the Common Language Effect Size.

Alternative Tests: `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Student. (1908). The probable error of a mean. *Biometrika, 6*(1), 1–25. https://doi.org/10.1093/biomet/6.1.1

### Examples

```
#Example 1: Numeric dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2['Gen_Age']
ts_student_t_os(ex1)
ts_student_t_os(ex1, mu=22)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
ts_student_t_os(ex2)
```

---

## `ts_student_t_ps` *Student t Test (Paired Samples)*

### Description

The assumption about the population (null hypothesis) for this test is a pre-defined difference between two means, usually zero (i.e. the difference between the (arithmetic) means is zero, they are the same in the population). If the p-value (significance) is then below a pre-defined threhold (usually 0.05), the assumption is rejected.

### Usage

```
ts_student_t_ps(field1, field2, dmu = 0)
```

### Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `dmu` | difference according to null hypothesis (default is 0) |

## Details

The formula used is:

$$t_p = \frac{\bar{d} - d_{H0}}{SE}$$

$$sig. = 2 \times \left(1 - \text{T}\left(\left|t_p\right|, df\right)\right)$$

With:

$$\bar{d} = \bar{x}_1 - \bar{x}_2$$

$$SE = \sqrt{\frac{\sigma_{s^2}}{n}}$$

$$s_d^2 = \frac{\sum_{i=1}^{n}\left(d_i - \bar{d}_i\right)^2}{n-1}$$

$$d_i = x_{i,1} - x_{i,2}$$

$$\bar{d} = \frac{\sum_{i=1}^{n} d_i}{n}$$

### Symbols used:

- $n$ the number of pairs (sample size)
- $x_{i,1}$ the i-th score of the first variable
- $x_{i,2}$ the i-th score of the second variable
- $d_{H0}$ the expected difference in the population
- $\text{T}\left(\ldots, \ldots\right)$ the cumulative distribution function of the Student t distribution

### Alternatives

R's *stats* library

t.test(var1, var2, paired=TRUE)

t.test(var1, var2, paired=TRUE, mu=5)

## Value

A dataframe with:

| | |
|---|---|
| `n` | the number of scores |
| `statistic` | the test statistic (t-value) |
| `df` | the degrees of freedom |
| `pValue` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Student. (1908). The probable error of a mean. *Biometrika, 6*(1), 1–25. doi:10.1093/biomet/6.1.1

---

`ts_trimmed_mean_is` *Independent Samples Trimmed/Yuen Mean Test*

---

## Description

A test to compare two means. The null hypothesis would be that the means of each category are equal in the population.

There are four similar tests, with different assumptions.

| test | equal variance | normality |
|---|---|---|
| Student | yes | yes |
| Welch | no | yes |
| Trimmed | yes | no |
| Yuen-Welch | no | no |

The Student and Welch are available as separate functions. The Trimmed Means and Yuen-Welch test are available in this one.

## Usage

```
ts_trimmed_mean_is(
  catField,
  scaleField,
  categories = NULL,
  dmu = 0,
  trimProp = 0.1,
  se = c("yuen", "wilcox")
)
```

## Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scaleField` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `dmu` | Optional difference according to null hypothesis (default is 0) |
| `trimProp` | Optional proportion to trim in total for each category. If for example set to 0.1 then 0.05 from each side for each category will be trimmed. Default is 0.1. |
| `se` | Optional to indicate which standard error to use. Either "yuen" (default) or "yuen-dixon". |

## Details

### YUEN

The default `se="yuen"` will perform a Yuen-Welch test.

The formula used is (Yuen, 1974, p. 167):

$$t = \frac{\bar{x}_{t,1} - \bar{x}_{t,2}}{SE}$$

$$sig = 2 \times \left(1 - T\left(\left|t\right|, df\right)\right)$$

With:

$$SE = \sqrt{\frac{s_{w,1}^2}{m_1} + \frac{s_{w,2}^2}{m_2}}$$

$$s_{w,i}^2 = \frac{SSD_{w,i}}{m_i - 1}$$

$$df = \frac{1}{\frac{c^2}{m_1 - 1} + \frac{(1-c)^2}{m_2 - 1}}$$

$$c = \frac{\frac{s_{w,1}^2}{m_1}}{\frac{s_{w,1}^2}{m_1} + \frac{s_{w,2}^2}{m_2}}$$

$$\bar{x}_{t,i} = \frac{\sum_{j=g_i+1}^{n_i-g_i} y_{i,j}}{}$$

$$g_i = \lfloor n_i \times p_t \rfloor$$

$$m_i = n\_2 \times g_i$$

$$SSD_{w,i} = g_i \times \left(y_{i,g_i+1} - \bar{x}_{wi}\right)^2 + g_i \times \left(y_{i,n_i-g_i} - \bar{x}_{w,i}\right)^2 + \sum_{j=g+1}^{n_i-g_i} \left(y_{i,j} - \bar{x}_{w,i}\right)^2$$

$$\bar{x}_{w,i} = \frac{\bar{x}_{t,i} \times m_i + g_i \times \left(y_{i,g_i+1} + y_{i,n_i-g_i}\right)}{n_i}$$

*Symbols used:*

- $x_{t,i}$ the trimmed mean of the scores in category i
- $x_{w,i}$ The Winsorized mean of the scores in category i
- $SSD_{w,i}$ the sum of squared deviations from the Winsorized mean of category i
- $m_i$ the number of scores in the trimmed data set from category i
- $y_{i,j}$ the j-th score after the scores in category i, after they are sorted from low to high
- $p_t$ the proportion of trimming on each side, we can define

### YUEN-DIXON

If `se="yuen-dixon` a trimmed means test will be performed.

The formula used is (Yuen & Dixon, 1973, p. 394):

$$t = \frac{\bar{x}_{t,1} - \bar{x}_{t,2}}{SE}$$

$$sig = 2 \times \left(1 - T\left(\left|t\right|, df\right)\right)$$

With:

$$SE = \sqrt{\frac{SSD_{w,1} + SSD_{w,2}}{m_1 + m_2 - 2} \times \left(\frac{1}{m_1} + \frac{1}{m_2}\right)}$$

$$df = m_1 + m_2 - 2$$

$$\bar{x}_{t,i} = \frac{\sum_{j=g_i+1}^{n_i-g_i} y_{i,j}}{}$$

$$g_i = \lfloor n_i \times p_t \rfloor$$

$$m_i = n\_2 \times g_i$$

$$SSD_{w,i} = g_i \times \left(y_{i,g_i+1} - \bar{x}_{w,i}\right)^2 + g_i \times \left(y_{i,n_i-g_i} - \bar{x}_{w,i}\right)^2 + \sum_{j=g+1}^{n_i-g_i} \left(y_{i,j} - \bar{x}_{w,i}\right)^2$$

$$\bar{x}_{w,i} = \frac{\bar{x}_{t,i} \times m_i + g_i \times \left(y_{i,g_i+1} + y_{i,n_i-g_i}\right)}{n_i}$$

*Symbols used:*

- $x_{ti}$ the trimmed mean of the scores in category i
- $x_{wi}$ The Winsorized mean of the scores in category i
- $SSD_{wi}$ the sum of squared deviations from the Winsorized mean of category i
- $m_i$ the number of scores in the trimmed data set from category i
- $y_{i,j}$ the j-th score after the scores in category i, after they are sorted from low to high
- $p_t$ the proportion of trimming on each side, we can define

## Value

A dataframe with:

| | |
|---|---|
| `n cat. 1` | the sample size of the first category |
| `n cat. 2` | the sample size of the second category |
| `trim mean cat. 1` | the sample trimmed mean of the first category |
| `trim mean cat. 2` | the sample trimmed mean of the second category |
| `diff.` | difference between the two sample means |
| `hyp. diff.` | hypothesized difference between the two population means |
| `statistic` | the test statistic (t-value) |
| `df` | degrees of freedom |
| `pValue` | the significance (p-value) |
| `test` | name of test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Yuen, K. K. (1974). The two-sample trimmed t for unequal population variances. *Biometrika, 61*(1), 165–170. https://doi.org/10.1093/biomet/61.1.165

Yuen, K. K., & Dixon, W. J. (1973). The approximate behaviour and performance of the two-sample trimmed t. *Biometrika, 60*(2), 369–374. https://doi.org/10.2307/2334550

---

## `ts_trimmed_mean_os` *One-Sample (Yuen or Yuen-Welch) Trimmed Mean Test*

---

### Description

A variation on a one-sample Student t-test where the data is first trimmed, and the Winsorized variance is used.

The assumption about the population for this test is that the mean in the population is equal to the provide mu value. The test will show the probability of the found test statistic, or more extreme, if this assumption would be true. If this is below a specific threshold (usually 0.05) the assumption is rejected.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_trimmed_mean_os(data, mu = NULL, trimProp = 0.1, se = "yuen")
```

### Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `mu` | optional hypothesized trimmed mean, otherwise the midrange will be used |
| `trimProp` | optional proportion to trim in total (half will be trimmed from each side) |
| `se` | optional method to use to determine standard error. Either "yuen" (default) or "wilcox" |

### Details

The formula used is:

$$\frac{\bar{x}_t - \mu_{H_0}}{SE}$$

$$sig = 2 \times \left(1 - T\left(\left|t\right|, df\right)\right)$$

With:

$$\bar{x}_t = \frac{\sum_{i=g+1}^{n-g} y_i}{}$$

$$g = \lfloor n \times p_t \rfloor$$

$$m = n - 2 \times g$$

$$SE = \sqrt{\frac{SSD_w}{m} \times (m - 1)}$$

or:

$$SE = \frac{\sqrt{\frac{SSD_w}{n-1}}}{(1 - 2 \times p_t) \times \sqrt{n}}$$

$$SSD_w = g \times \left(y_{g+1} - \bar{x}_w\right)^2 + g \times \left(y_{n-g} - \bar{x}_w\right)^2 + \sum_{i=g+1}^{n-g} \left(y_i - \bar{x}_w\right)^2$$

$$\bar{x}_w = \frac{\bar{x}_t \times m + g \times \left(y_{g+1} + y_{n-g}\right)}{n}$$

*Symbols used:*

- $x_t$ the trimmed mean of the scores
- $x_w$ The Winsorized mean
- $SSD_w$ the sum of squared deviations from the Winsorized mean
- $m$ the number of scores in the trimmed data set from category i
- $y_i$ the i-th score after the scores are sorted from low to high
- $p$ the proportion of trimming on each side, we can define

The test is often also referred to as a Yuen test, or Yuen-Welch test.

The standard error can either be calculated using the first SE, which for example can be found in Tukey and McLaughlin (1963, p. 342), and seems similar to the independent samples version of this test as proposed by Yuen (1974, p. 167)

The second version is used in the other libraries, and can be found in Wilcox (2012, p. 157), or Peró-Cebollero and Guàrdia-Olmos (2013, p. 409).

## Value

A dataframe with:

| | |
|---|---|
| `trim. mean` | the sample trimmed mean |
| `mu` | hypothesized trimmed mean |
| `SE` | the standard error |
| `statistic` | test statistic |
| `df` | degrees of freedom |
| `p-value` | p-value (sig.) |
| `test used` | test used |

## Before, After and Alternatives

Before this you might want to create a binned frequency table or a visualisation: `tab_frequency_bins`, to create a binned frequency table. `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

After this you might want an effect size measure: `es_cohen_d_os`, for for Cohen d'. `es_hedges_g_os`, for Hedges g. `es_common_language_os`, for the Common Language Effect Size.

Alternative Tests: `ts_student_t_os`, for One-Sample Student t-Test. `ts_z_os`, for One-Sample Z Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Peró-Cebollero, M., & Guàrdia-Olmos, J. (2013). The adequacy of different robust statistical tests in comparing two independent groups. *Psicológica*, 34, 407–424.

Tukey, J. W., & McLaughlin, D. H. (1963). Less vulnerable confidence and significance procedures for location based on a single sample: Trimming/Winsorization 1. *Sankhyā: The Indian Journal of Statistics, 25*(3), 331–352.

Wilcox, R. R. (2012). *Introduction to robust estimation and hypothesis testing* (3rd ed.). Academic Press.

Yuen, K. K. (1974). The two-sample trimmed t for unequal population variances. *Biometrika, 61*(1), 165–170. doi:10.1093/biomet/61.1.165

### Examples

```
#Example 1: Numeric dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c(””, ”NA”))
#Example 1: Numeric dataframe
ex1 = df2['Gen_Age']
ts_trimmed_mean_os(ex1)
ts_trimmed_mean_os(ex1, mu=23, trimProp=0.15, se=”wilcox”)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
ts_trimmed_mean_os(ex2, trimProp=0.05)
```

---

## `ts_trinomial_os` *One-Sample Trinomial Test*

---

### Description

A test that could be used with ordinal data that includes ties

Similar as a sign-test but instead of ignoring scores that are tied with the hypothesized median they get included, hence instead of the binomial distribution, this will use the trinomial distribution.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com.

### Usage

```
ts_trinomial_os(data, levels = NULL, mu = NULL)
```

### Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `levels` | optional list to indicate what values represent |
| `mu` | optional hypothesized median, otherwise the midrange will be used |

### Details

The p-value is calculated using (Bian et al., 2009, p. 6):

$$p = 2 \times \sum_{i=n_d}^{n} \sum_{j=0}^{\lfloor \frac{n-i}{2} \rfloor} \text{tri}\left(\left(j, j+i, n-i\right), \left(p_{pos}, p_{neg}, p_0\right)\right)$$

With:

$$p_0 = \frac{n_0}{n}$$

$$p_{pos} = p_{neg} = \frac{1 - p_0}{n}$$

$$\left|n_{pos} - n_{neg}\right|$$

*Symbols used:*

- $n_0$ the number of scores equal to the hypothesized median
- $n_{pos}$ the number of scores above the hypothesized median
- $n_{neg}$ the number of scores below the hypothesized median
- $p_0$ the probability of the a score in the sample being equal to the hypothesized median
- $p_{pos}$ the population proportion of a score being above the hypothesized median
- $p_{neg}$ the population proportion of a score being below the hypothesized median
- $\text{tri}\left(\dots,\dots\right)$ the trinomial probability mass function

The paired version of the test is described in Bian et al. (1941), while Zaiontz (n.d.) mentions it can also be used for one-sample situations.

## Value

A dataframe with:

| | |
|---|---|
| `mu` | he hypothesized median |
| `n-pos` | the number scores above mu |
| `n-neg` | the number scores below mu |
| `n-tied` | the number of scores tied with mu |
| `p-value` | significance (p-value) |
| `test` | description of the test used |

## Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want to determine an effect size measure: `es_common_language_os`, for the Common Language Effect Size. `es_dominance`, for the Dominance score. `r_rank_biserial_os`, for the Rank-Biserial Correlation

Alternative tests: `ts_sign_os`, for One-Sample Sign Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Bian, G., McAleer, M., & Wong, W.-K. (2009). A trinomial test for paired data when there are many ties. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.1410589

Zaiontz, C. (n.d.). Trinomial test. Real Statistics Using Excel. Retrieved March 2, 2023, from https://real-statistics.com/non-parametric-tests/trinomial-test/

## Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Dataframe
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
ts_trinomial_os(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
ts_trinomial_os(ex2)
```

---

# `ts_trinomial_ps` *Trinomial Test (Paired Samples)*

---

## Description

A similar test as the sign test, but also includes the pairs that are tied.

## Usage

```
ts_trinomial_ps(field1, field2, levels = NULL, dmu = 0)
```

## Arguments

| | |
|---|---|
| `field1` | the numeric scores of the first variable |
| `field2` | the numeric scores of the second variable |
| `levels` | vector, optional. the levels from field1 and field2 |
| `dmu` | float, optional. The difference according to the null hypothesis (default is 0) |

## Details

The formula used (Bian et al., 2009, p. 6):

$$sig. = 2 \times \text{TRI}\left(\left(n_{pos}, n_{neg}, n_0\right), \left(p_{pos}, p_{neg}, p_0\right)\right)$$

With:

$$n_{pos} = \sum_{i=1}^{n} \begin{cases} 1 & \text{if } d_i > d_{H0} \\ 0 & \text{if } d_i \leq d_{H0} \end{cases}$$

$$n_{neg} = \sum_{i=1}^{n} \begin{cases} 0 & \text{if } d_i \geq d_{H0} \\ 1 & \text{if } d_i < d_{H0} \end{cases}$$

$$n_0 = \sum_{i=1}^{n} \begin{cases} 1 & \text{if } d_i = d_{H0} \\ 0 & \text{if } d_i \neq d_{H0} \end{cases}$$

$$d_i = x_i - y_i$$

$$p_0 = \frac{n_0}{n}$$

$$p_{pos} = p_{neg} = \frac{1 - p_0}{2}$$

The cumulative mass function of the trinomial distribution is then calculated using:

$$\text{TRI}\left(\left(n_{pos}, n_{neg}, n_0\right), \left(p_{pos}, p_{neg}, p_0\right)\right) = \sum_{i=n_d}^{n} \sum_{j=0}^{\left\lfloor \frac{n-i}{2} \right\rfloor} \text{tri}\left(\left(j, j+i, n-j-(j+i)\right), \left(p_{pos}, p_{neg}, p_0\right)\right)$$

$$n_d = \left|n_{pos} - n_{neg}\right|$$

The probability mass function of the trinomial distribution is (Bian et al., 2009, p. 5):

$$\text{tri} = \left(\left(n_a, n_b, n_c\right), \left(p_a, p_b, p_c\right)\right) = \frac{n!}{a! \times b! \times c!} \times p_a^{n_a} \times p_b^{n_b} \times p_c^{n_c}$$

*Symbols used:*

- $n$ is the number of pairs with a difference unequal to zero
- $n_{pos}$ the number of pairs with a difference greater than the null hypothesis
- $n_{neg}$ the number of pairs with a difference greater than the null hypothesis
- $n_0$ the number of pairs with no difference with the null hypothesis
- $d_{H0}$ the difference according to the null hypothesis, usually 0
- $x_i$ the i-th score from the first variable
- $y_i$ the i-th score from the second variable
- $\text{tri}\left(\ldots, \ldots\right)$ the probability mass function of the trinomial distribution

**Alternatives**

*library(EMT)*

datFrame = na.omit(data.frame(ord1,ord2))

d = datFrame$ord1 - datFrame$ord2

pos = sum(d>0)

neg = sum(d<0)

ties = sum(d==0)

n = pos + neg + ties

p0 = ties/n

p1 = (1 - p0)/2

multinomial.test(c(pos, neg, ties), c(p1, p1, p0))

## Value

A dataframe with:

| | |
|---|---|
| `n pos` | the number of scores with a positive difference |
| `n neg` | the number of scores with a negative difference |
| `n 0` | the number of scores with a no difference |
| `p-Value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Bian, G., McAleer, M., & Wong, W.-K. (2009). A trinomial test for paired data when there are many ties. *SSRN Electronic Journal*. doi:10.2139/ssrn.1410589

---

# `ts_wald_os` *One-Sample Wald Test*

## Description

A one-sample score test could be used with binary data, to test if the two categories have a significantly different proportion. It is an approximation of a binomial test, by using a standard normal distribution. Since the binomial distribution is discrete while the normal is continuous, a so-called continuity correction can (should?) be applied.

The null hypothesis is usually that the proportions of the two categories in the population are equal (i.e. 0.5 for each). If the p-value of the test is below the pre-defined alpha level (usually 5% = 0.05) the null hypothesis is rejected and the two categories differ in proportion significantly.

The input for the function doesn't have to be a binary variable. A nominal variable can also be used and the two categories to compare indicated.

A significance in general is the probability of a result as in the sample, or more extreme, if the null hypothesis is true.

Some info on the different tests can be found in video. This function is shown in this YouTube video and the test is also described at PeterStatistics.com

## Usage

```
ts_wald_os(data, p0 = 0.5, p0Cat = NULL, codes = NULL, cc = c("none", "yates"))
```

## Arguments

| | |
|---|---|
| `data` | A vector with the data |
| `p0` | Optional hypothesized proportion for the first category (default is 0.5) |
| `p0Cat` | Optional the category for which p0 was used |
| `codes` | Optional vector with the two codes to use |
| `cc` | use of continuity correction (default is "none") |

## Details

To decide on which category is associated with p0 the following is used:

- If codes are provided, the first code is assumed to be the category for the p0.
- If p0Cat is specified that will be used for p0 and all other categories will be considered as category 2, this means if there are more than two categories the remaining two or more (besides p0Cat) will be merged as one large category.

- If neither codes or p0Cat is specified and more than two categories are in the data a warning is printed and no results.
- If neither codes or p0Cat is specified and there are two categories, p0 is assumed to be for the category closest matching the p0 value (i.e. if p0 is above 0.5 the category with the highest count is assumed to be used for p0)

This test differs from the one-sample score test in the calculation of the standard error. For the 'regular' version this is based on the expected proportion, while for the Wald version it is done with the observed proportion.

The formula used (Wald, 1943):

$$z = \frac{x - \mu}{SE}$$

With:

$$\mu = n \times p_0$$

$$SE = \sqrt{x \times \left(1 - \frac{x}{n}\right)}$$

*Symbols used:*

- $x$ is the number of successes in the sample
- $p_0$ the expected proportion (i.e. the proportion according to the null hypothesis)

If the Yates continuity correction is used the formula changes to (Yates, 1934, p. 222):

$$z_{Yates} = \frac{|x - \mu| - 0.5}{SE}$$

The formula used in the calculation is the one from IBM (2021, p. 997). IBM refers to Agresti, most likely Agresti (2013, p. 10), who in turn refer to Wald (1943)

## Value

Dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the test value |
| `pValue` | two-sided p-value |
| `test` | a description of the test used |

## Before, After and Alternatives

Before running the test you might first want to get an impression using a frequency table: tab_frequency

After the test you might want an effect size measure: `es_cohen_g`, for Cohen g `es_cohen_h_os`, for Cohen h' `es_alt_ratio`, for Alternative Ratio

Alternatives for this test could be: `ts_binomial_os`, for One-Sample Binomial Test `ts_score_os`, for One-Sample Score Test

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Agresti, A. (2013). *Categorical data analysis* (3rd ed.). Wiley.

IBM SPSS Statistics Algorithms. (2021). IBM.

Wald, A. (1943). Tests of statistical hypotheses concerning several parameters when the number of observations is large. *Transactions of the American Mathematical Society, 54*(3), 426–482. doi:10.2307/1990256

Yates, F. (1934). Contingency tables involving small numbers and the chi square test. *Supplement to the Journal of the Royal Statistical Society, 1*(2), 217–235. doi:10.2307/2983604

## Examples

```
#Example 1: Numeric list
ex1 = c(1, 1, 2, 1, 2, 1, 2, 1)
ts_wald_os(ex1)
ts_wald_os(ex1, p0=0.3)
ts_wald_os(ex1, p0=0.3, cc="yates")

#Example 2: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ts_wald_os(df1['sex'])
ts_wald_os(df1['mar1'], codes=c("DIVORCED", "NEVER MARRIED"))
```

---

# `ts_welch_owa` *Welch One-Way ANOVA*

## Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

Delacre et al. (2019) recommend to use the Welch ANOVA instead of the classic and Brown-Forsythe versions, but there are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

## Usage

```
ts_welch_owa(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

**Details**

The formula used is (Welch, 1951, pp. 334-335):

$$F_w = \frac{\frac{1}{k-1}\times\sum_{j=1}^{k} w_j\times\left(\bar{x}_j-\bar{y}_w\right)^2}{1+2\times\lambda\times\frac{k-2}{k^2-1}}$$

$$df_1 = k-1$$

$$df_2 = \frac{k^2-1}{3\times\lambda}$$

$$sig. = 1-F\left(F_W, df_1, df_2\right)$$

With:

$$\lambda = \sum_{j=1}^{k}\frac{\left(1-h_j\right)^2}{n_j-1}$$

$$\bar{y}_w = \frac{\sum_{j=1}^{k} w_j\times\bar{x}_j}{\sum_{j=1}^{k} w_j} = \sum_{j=1}^{k} h_j\times\bar{x}_j$$

$$h_j = \frac{w_j}{w}$$

$$w = \sum_{j=1}^{k} w_j$$

$$w_j = \frac{n_j}{s_j^2}$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j}-\bar{x}_j\right)^2}{n_j-1}$$

$$\bar{x}_j = \frac{\sum_{i=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $x_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $w_j$ the weight for category j
- $h_j$ the adjusted weight for category j
- $df_i$ the i-th degrees of freedom

The formula can also be written as:

$$F_W = \frac{\chi_{Cochran}^2}{k-1+2\times\lambda\times\frac{k-2}{k+1}}$$

Where $\chi_{Cochran}^2$ is the test statistic of the Cochran one-way test

Cavus and Yazici (2020) make a difference between the Welch and the Welch-Aspin ANOVA. The only difference in the article is that with the Welch 2×(k-2) is used, while in the Welch-Aspin version 2×k-2. I think this is a mistake in their formula, since the article they refer to from Aspin is about two means.

Johansen F test (Johansen, 1980) will give the same results

### Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the test statistic (F value) |
| `df1` | the degrees of freedom |
| `df2` | the degrees of freedom |
| `pValue` | the significance (p-value) |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cavus, M., & Yazıcı, B. (2020). Testing the equality of normal distributed and independent groups' means under unequal variances by doex package. *The R Journal, 12*(2), 134. doi:10.32614/RJ-2021-008

Delacre, M., Leys, C., Mora, Y. L., & Lakens, D. (2019). Taking parametric assumptions seriously: Arguments for the use of Welch's F-test instead of the classical F-test in one-way ANOVA. *International Review of Social Psychology, 32*(1), 1–12. doi:10.5334/irsp.198

Johansen, S. (1980). The Welch-James approximation to the distribution of the residual sum of squares in a weighted linear regression. *Biometrika, 67*(1), 85–92. doi:10.1093/biomet/67.1.85

Welch, B. L. (1947). The generalization of 'Student's' problem when several different population variances are involved. *Biometrika, 34*(1/2), 28–35. doi:10.2307/2332510

Welch, B. L. (1951). On the comparison of several mean values: An alternative approach. *Biometrika, 38*(3/4), 330–336. doi:10.2307/2332579

---

## `ts_welch_t_is` *Welch t Test (Independent Samples)*

### Description

A test to compare two means. The null hypothesis would be that the means of each category are equal in the population.

Unlike the Student t-test, the Welch test does not assume the variances of the two categories to be equal in the population. Ruxten (2006) even argues that the Welch t-test should always be prefered over the Student t-test.

There are four similar tests, with different assumptions.

| test | equal variance | normality |
|---|---|---|
| Student | yes | yes |
| Welch | no | yes |
| Trimmed | yes | no |
| Yuen-Welch | no | no |

The Trimmed and Yuen-Welch can be found in the **ts_trimmed_mean_is**(), and the Student t-test with the **ts_student_t_is().**

## Usage

```
ts_welch_t_is(catField, scaleField, categories = NULL, dmu = 0)
```

## Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scaleField` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `dmu` | Optional difference according to null hypothesis (default is 0) |

## Details

The formula used is:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{SE}$$

$$df = \frac{SE^4}{\frac{\left(s_1^2\right)^2}{n_1^2\times\left(n_1 - 1\right)} + \frac{\left(s_2^2\right)^2}{n_2^2\times\left(n_2 - 1\right)}}$$

$$sig. = 2\times\left(1 - T\left(\left|t\right|, df\right)\right)$$

With:

$$SE = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

$$s_i^2 = \frac{\sum_{j=1}^{n_i}\left(x_{i,j} - \bar{x}_i\right)^2}{n_i - 1}$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{i,j}}{n_i}$$

*Symbols used:*

- $x_{i,j}$ the j-th score in category i
- $n_i$ the number of scores in category i

## Value

A dataframe with:

| | |
|---|---|
| `n cat. 1` | the sample size of the first category |
| `n cat. 2` | the sample size of the second category |
| `mean cat. 1` | the sample mean of the first category |
| `mean cat. 2` | the sample mean of the second category |
| `diff.` | difference between the two sample means |
| `hyp. diff.` | hypothesized difference between the two population means |
| `statistic` | the test statistic (t-value) |
| `df` | degrees of freedom |
| `pValue` | the significance (p-value) |
| `test` | name of test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Ruxton, G. D. (2006). The unequal variance t-test is an underused alternative to Student's t-test and the Mann–Whitney U test. *Behavioral Ecology, 17*(4), 688–690. https://doi.org/10.1093/beheco/ark016

Welch, B. L. (1947). The generalization of 'Student's' problem when several different population variances are involved. *Biometrika, 34*(1/2), 28–35. https://doi.org/10.2307/2332510

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['age']
ex1 = replace(ex1, ex1=="89 OR OLDER", "90")
ts_welch_t_is(df1['sex'], ex1)

#Example 2: vectors
scores = c(20,50,80,15,40,85,30,45,70,60, NA, 90,25,40,70,65, NA, 70,98,40)
groups = c("nat.","int.","int.","nat.","int.", "int.","nat.","nat.","int.",
"int.","int.","int.","int.","int.","nat.", "int." ,NA,"nat.","int.","int.")
ts_welch_t_is(groups, scores)
```

---

# `ts_wilcoxon_os` *One-Sample Wilcoxon Signed Rank Test*

## Description

The one-sample Wilcoxon signed rank test is often considered the non-parametric version of a one-sample t-test. It can be used to determine if the median is significantly different from an hypothesized value. It actually doesn't always tests this specifically, but more if the mean rank is significantly different.

If the p-value is the probability of a result as in the sample, or more extreme, if the assumption about the population would be true. If this is below a certain threshold (usually 0.05) the assumption about the population is rejected. For this test the assumed median for the population is then incorrect.

Results in software packages for this test can vary, since there are a few different approaches. Especially if there are so-called ties. See the details for more information.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com.

## Usage

```
ts_wilcoxon_os(
  data,
  levels = NULL,
  mu = NULL,
  ties = TRUE,
  appr = c("wilcoxon", "exact", "imanz", "imant"),
  eqMed = c("wilcoxon", "zsplit", "pratt"),
```

```
  cc = FALSE
)
```

## Arguments

| | |
|---|---|
| `data` | dataframe with scores as numbers, or if text also provide levels |
| `levels` | optional vector with levels in order |
| `mu` | optional hypothesized median, otherwise the midrange will be used |
| `ties` | optional boolean to use a tie correction (default is True) |
| `appr` | optional which method to use for approximation (default is "wilcoxon") |
| `eqMed` | optional method to deal with scores equal to hypMed (default is "wilcoxon") |
| `cc` | optional boolean to use a continuity correction (default is FALSE) |

## Details

The unadjusted test statistic is given by:

$$W = \sum_{i=1}^{n_r^+} r_i^+$$

With:

$$r = \text{rank}(|d|)$$

$$d_i = y_i - \theta$$

*Symbols used:*

- $n_r^+$ is the number of ranks with a positive deviation from the hypothesized median
- $r_i^+$ the i-th rank of the ranks with a positive deviation from the hypothesized median
- $\theta$ is the median tested (the hypothesized median).
- $y_i$ is the i-th score of the variable after removing scores that were equal to $\theta$

If there are no ties, an exact method can be used, using the Sign Rank Distribution. R has this available with *psignrank()*. The exact test can be found in Zaiontz (n.d.)

### Approximations

If the sample size is large enough, we can use a normal approximation. What is large enough varies quite per author. A few examples: n > 8 (slideplayer, 2015), n > 15 (SigMaxl, n.d.), n > 20 (Wikipedia, n.d.), n > 25 (Harris & Hardin, 2013), n > 30 (Winthrop, n.d.) .

The z-statistic is given by (appr="wilcoxon", ties=FALSE, cc=FALSE):

$$Z = \frac{W - \mu_w}{\sigma_w}$$

or with a ties correction (appr="wilcoxon", ties=TRUE, cc=FALSE):

$$Z_{adj} = \frac{W - \mu_w}{\sigma_w^*}$$

With:

$$\mu_w = \frac{n_r \times (n_r + 1)}{4}$$

$$\sigma_w^2 = \frac{n_r \times (n_r + 1) \times (2 \times n_r + 1)}{24}$$

$$\sigma_w^{*2} = \sigma_w^2 - A$$

$$A = \frac{\sum_{i=1}^{k} \left(t_i^3 - t_i\right)}{48}$$

*Additional symbols used*

- $n_r$ is the number of ranks used
- $k$ the number of unique ranks
- $t_i$ the frequency of the i-th unique rank

A Yates continuity correction can simply be applied: In case of no ties (appr="wilcoxon", ties=FALSE, cc=TRUE):

$$Z = \frac{|W - \mu_w| - 0.5}{\sigma_w}$$

In case of ties (appr="wilcoxon", ties=TRUE, cc=TRUE):

$$Z_{adj} = \frac{|W - \mu_w| - 0.5}{\sigma_w^*}$$

An alternative approximation using the Student t distribution is given by Iman (1974, p. 799). The formula is (appr="imant", ties=FALSE, cc=FALSE):

$$t = \frac{W - \mu_w}{\sqrt{\frac{\sigma_w^2 \times n_r - (W - \mu_w)^2}{n_r - 1}}}$$

or with the ties correction (appr="imant", ties=TRUE, cc=FALSE):

$$t = \frac{W - \mu_w}{\sqrt{\frac{\sigma_w^{*2} \times n_r - (W - \mu_w)^2}{n_r - 1}}}$$

The two versions for with a continuity correction are: No ties correction, but continuity (appr="imant", ties=FALSE, cc=TRUE):

$$t = \frac{|W - \mu_w| - 0.5}{\sqrt{\frac{\sigma_w^2 \times n_r - (|W - \mu_w| - 0.5)^2}{n_r - 1}}}$$

Both corrections (appr="imant", ties=TRUE, cc=TRUE):

$$t = \frac{|W - \mu_w| - 0.5}{\sqrt{\frac{\sigma_w^{*2} \times n_r - (|W - \mu_w| - 0.5)^2}{n_r - 1}}}$$

Iman (1974, p. 803) also provides a combination of the t-approximation and the regular z-approximation The equation is given by (appr="imanz"):

$$Z_I = \frac{Z}{2} \times \left(1 + \sqrt{\frac{n_r - 1}{n_r - Z^2}}\right)$$

The $Z$ is any of the previous methods.

**Ties with mu**

The default (eqMed="wilcoxon") removes first any scores that are equal to the hypothesized median. There are two alternative methods for this.Both re-define $d_i$ to:

$$d_i = x_i - \theta$$

Where $x_i$ is simply the i-th score.

For the z-split method we only need to re-define:

$$W = \frac{\sum_{i=1}^{n_{d_0}} r_{i,0}}{2} + \sum_{i=1}^{n_r^+} r_i^+$$

Where $n_{d_0}$ is the number of scores that equal the hypothesized median, and $r_{i,0}$ is the rank of the i-th score that equals the hypothesized median.

In essence we added half the sum of the ranks that were equal to the hypothesized median.

For the z-split method all other calculations than go the same.

For the Pratt (1959) method we also re-define:

$$\mu_w = \frac{n_r \times (n_r + 1) - n_{d_0} \times (n_{d_0} + 1)}{4}$$

$$\sigma_w^2 = \frac{n_r \times (n_r + 1) \times (2 \times n_r + 1) - n_{d_0} \times (n_{d_0} + 1) \times (2 \times n_{d_0} + 1)}{24}$$

For the Pratt method, the ties correction still excludes the ties for the scores that equal the hypothesized median, but for the z-split method it will include them.

For both methods now $n_r = n$, where n is the number of scores.

The Pratt (1959) method and z-split method were found in Python's documentation for scipy's Wilcoxon function (scipy, n.d.). They also refer to Cureton (1967) for the Pratt method.

## Value

A dataframe with:

| | |
|---|---|
| `mu` | the hypothesized median according to the null |
| `W` | the Wilcoxon W value |
| `statistic` | test statistic |
| `df` | degrees of freedom (only applicable for Iman t approximation) |
| `pValue` | significance (p-value) |
| `testUsed` | description of the test used |

## Before, After and Alternatives

Before this measure you might want an impression using a frequency table or a visualisation: `tab_frequency`, for a frequency table `vi_bar_stacked_single`, or Single Stacked Bar-Chart. `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

After this you might want to determine an effect size measure: `es_common_language_os`, for the Common Language Effect Size. `es_dominance`, for the Dominance score. `r_rank_biserial_os`, for the Rank-Biserial Correlation

Alternative tests: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Cureton, E. E. (1967). The normal approximation to the signed-rank sampling distribution when zero differences are present. *Journal of the American Statistical Association, 62*(319), 1068–1069. https://doi.org/10.1080/01621459.1967.10500917

Harris, T., & Hardin, J. W. (2013). Exact Wilcoxon Signed-Rank and Wilcoxon Mann–Whitney Ranksum Tests. *The Stata Journal, 13*(2), 337–343. https://doi.org/10.1177/1536867X1301300208

Iman, R. L. (1974). Use of a t-statistic as an approximation to the exact distribution of the wildcoxon signed ranks test statistic. *Communications in Statistics, 3*(8), 795–806. https://doi.org/10.1080/03610927408827178

Pratt, J. W. (1959). Remarks on zeros and ties in the Wilcoxon signed rank procedures. *Journal of the American Statistical Association, 54*(287), 655–667. https://doi.org/10.1080/01621459.1959.10501526

scipy. (n.d.). Scipy.stats.wilcoxon. Scipy. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.wilcoxon.html

SigMaxl. (n.d.). One Sample Wilcoxon Sign Test Exact. Retrieved August 30, 2020, from https://www.sigmaxl.com/OneSampleSignWilcoxonExact.shtml

slideplayer. (2015, June 13). Using statistics to make inferences 6.

Wikipedia. (n.d.). Wilcoxon signed-rank test. In Wikipedia. Retrieved August 30, 2020, from https://en.wikipedia.org/w/index.php?title=Wilcoxon_signed-rank_test&oldid=974561084

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin, 1*(6), 80. https://doi.org/10.2307/3001968

Winthrop. (n.d.). The Wilcoxon signed rank test for one sample. Winthrop Univerisy Hospital. https://nyuwinthrop.org/wp-content/uploads/2019/08/wilcoxon-sign-rank-test-one-sample.pdf

Zaiontz, C. (n.d.). Wilcoxon signed ranks exact test. Real Statistics Using Excel. Retrieved January 25, 2023, from https://real-statistics.com/non-parametric-tests/wilcoxon-signed-ranks-test/wilcoxon-signed-ranks-exact-test/

## Examples

```
#Example 1: Text dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
ts_wilcoxon_os(ex1, levels=order)

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
ts_wilcoxon_os(ex2)
```

---

## `ts_wilcoxon_ps` *Paired Samples Wilcoxon Signed Rank Test*

---

### Description

The paired-sample Wilcoxon signed rank test is often considered the non-parametric version of a paired-samples t-test. It can be used to determine if the median is significantly different between the two variables. It actually doesn't always tests this specifically, but more if the mean rank is significantly different.

If the p-value is the probability of a result as in the sample, or more extreme, if the assumption about the population would be true. If this is below a certain threshold (usually 0.05) the assumption about the population is rejected.

Results in software packages for this test can vary, since there are a few different approaches. Especially if there are so-called ties.

This function simply determines the differences between the two provided variables, and then passes these differences along to the one-sample version. See ts_wilcoxon_os() for details on this.

### Usage

```
ts_wilcoxon_ps(
  field1,
  field2,
  levels = NULL,
  dmu = 0,
  appr = c("wilcoxon", "exact", "imant", "imanz"),
  noDiff = c("wilcoxon", "zsplit", "pratt"),
  ties = TRUE,
  cc = FALSE
)
```

### Arguments

| | |
|---|---|
| `field1` | the numeric scores of the first variable |
| `field2` | the numeric scores of the second variable |
| `levels` | vector, optional. the levels from field1 and field2 |
| `dmu` | float, optional. The difference according to the null hypothesis (default is 0) |
| `appr` | c("wilcoxon", "exact", "imanz", "imant") optional which method to use for approximation (default is "wilcoxon") |
| `noDiff` | c("wilcoxon", "pratt", "zsplit") optional method to deal with scores equal on both variables (default is "wilcoxon") |
| `ties` | optional boolean to use a tie correction (default is True) |
| `cc` | optional boolean to use a continuity correction (default is False) |

## Details

The unadjusted test statistic is given by:

$$W = \min\left(W_{neg}, W_{pos}\right)$$

With:

$$W_{pos} = \sum_{i=1}^{n} \begin{cases} r_i & \text{if } d_i > 0 \\ 0 & \text{if } d_i \leq 0 \end{cases}$$

$$W_{neg} = \sum_{i=1}^{n} \begin{cases} r_i & \text{if } d_i < 0 \\ 0 & \text{if } d_i \geq 0 \end{cases}$$

$$d_i = x_i - y_i$$

*Symbols used:*

- $n$ the number of scores (equal for each variable)
- $x_i$ the i-th score on the first variable
- $y_i$ the i-th score on the second variable
- $r_i$ the i-th rank of the absolute differences ($d_i$)
- $W_{pos}$ is the number of ranks with a positive difference
- $W_{neg}$ is the number of ranks with a negative difference

The distribution and test for $W$ can now be performed the same way as for the one-sample case. See *ts_wilcoxon_os()* for details on the calculations. The $d_i$ scores are now the one-sample, and the hypothesized median would be 0.

**Alternatives**

*R's stats library *

wilcox.test(ord1, ord2, paired=TRUE, exact=FALSE, correct=TRUE)

wilcox.test(ord1, ord2, paired=TRUE, exact=FALSE, correct=FALSE)

*library(coin)*

wilcoxsign_test(ord1 ~ ord2, zero.method = "Wilcoxon")

wilcoxsign_test(ord1 ~ ord2, zero.method = "Pratt")

## Value

A dataframe with:

| | |
|---|---|
| `nr` | the number of ranks used in calculation |
| `mu` | the median according to the null hypothesis |
| `W` | the Wilcoxon W |
| `statistic` | the test statistic |
| `df` | degrees of freedom (only applicable for Iman t approximation) |
| `p-value` | the significance (p-value) |
| `test` | description of the test used |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin, 1*(6), 80. doi:10.2307/3001968

---

`ts_wilcox_owa` *Wilcox Test*

---

## Description

Tests if the means (averages) of each category could be the same in the population.

If the p-value is below a pre-defined threshold (usually 0.05), the null hypothesis is rejected, and there are then at least two categories who will have a different mean on the scaleField score in the population.

There are quite some alternatives for this, the stikpet library has Fisher, Welch, James, Box, Scott-Smith, Brown-Forsythe, Alexander-Govern, Mehrotra modified Brown-Forsythe, Hartung-Agac-Makabi, Özdemir-Kurt and Wilcox as options. See the notes from ts_fisher_owa() for some discussion on the differences.

## Usage

```
ts_wilcox_owa(nomField, scaleField, categories = NULL)
```

## Arguments

| | |
|---|---|
| `nomField` | the groups variable |
| `scaleField` | the numeric scores variable |
| `categories` | vector, optional. the categories to use from catField |

## Details

The formula used (Wilcox, 1988, pp. 110-111)

$$H = \frac{\sum_{j=1}^{k}\left(W_j - \bar{W}\right)^2}{\hat{\theta}}$$

$$df = k - 1$$

$$sig. = 1 - \chi^2\left(H, df\right)$$

With:

$$W_j = b_j \times x_{n_j,j} + \frac{1 - b_j}{n_j} \times \sum_{i=1}^{n_j-1} x_{i,j}$$

$$\bar{W} = \frac{\sum_{j=1}^{k} W_j}{k}$$

$$b_j = \frac{1 + \sqrt{\frac{(n_j-1)\times\left(n_j\times\hat{\theta}\right)}{s_j^2}}}{n_j}$$

$$\hat{\theta} = \max\left\{\frac{s_1^2}{n_1}, \frac{s_2^2}{n_2}, \ldots, \frac{s_k^2}{n_k}\right\}$$

$$s_j^2 = \frac{\sum_{i=1}^{n_j}\left(x_{i,j} - \bar{x}_j\right)^2}{n_j - 1}$$

$$\bar{x}_j = \frac{\sum_{j=1}^{n_j} x_{i,j}}{n_j}$$

*Symbols used:*

- $x_{i,j}$ the i-th score in category j
- $k$ the number of categories
- $n_j$ the sample size of category j
- $\bar{x}_j$ the sample mean of category j
- $s_j^2$ the sample variance of the scores in category j
- $df$ the degrees of freedom
- $\chi^2\left(\dots\right)$ the cumulative density function of the chi-square distribution

The original article has an error in the formula for $b_j$. There are missing brackets. Using the population version in the article of $c_j$ the formula used here was adapted.

## Value

A dataframe with:

| | |
|---|---|
| `n` | the sample size |
| `statistic` | the test statistic (chi-square value value) |
| `df` | the degrees of freedom |
| `p-value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Wilcox, R. R. (1988). A new alternative to the ANOVA F and new results on James's second-order method. *British Journal of Mathematical and Statistical Psychology, 41*(1), 109–117. https://doi.org/10.1111/j.2044-8317.1988.tb00890.x

---

# `ts_z_is` *Independent Samples Z Test*

## Description

A test to compare two means. It requires the population variances, but if these are unknown for large enough sample sizes, the sample variances can be used instead.

For smaller sample sizes a t-test (Student, Welch or Trimmed Means) could be used instead.

### Usage

```
ts_z_is(
  catField,
  scaleField,
  categories = NULL,
  dmu = 0,
  sigma1 = NULL,
  sigma2 = NULL
)
```

### Arguments

| | |
|---|---|
| `catField` | A vector with the categorical data |
| `scaleField` | A vector with the scores |
| `categories` | Optional to indicate which two categories of catField to use, otherwise first two found will be used. |
| `dmu` | Optional difference according to null hypothesis (default is 0) |
| `sigma1` | Optional population standard deviation of the first group, if NULL sample results will be used |
| `sigma2` | Optional population standard deviation of the second group, if NULL sample results will be used |

### Details

The formula used is:

$$z = \frac{\bar{x}_1 - \bar{x}_2}{SE}$$

$$sig. = 2\times\left(1 - \Phi\left(\left|z\right|\right)\right)$$

With:

$$SE = \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}$$

$$\sigma_i^2 \approx s_i^2 = \frac{\sum_{j=1}^{n_i}\left(x_{i,j} - \bar{x}_i\right)^2}{n_i - 1}$$

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{i,j}}{n_i}$$

*Symbols used:*

- $x_{i,j}$ the j-th score in category i
- $n_i$ the number of scores in category i

### Value

A dataframe with:

| | |
|---|---|
| `n cat. 1` | the sample size of the first category |
| `n cat. 2` | the sample size of the second category |
| `mean cat. 1` | the sample mean of the first category |

| | |
|---|---|
| `mean cat. 2` | the sample mean of the second category |
| `diff.` | difference between the two sample means |
| `hyp. diff.` | hypothesized difference between the two population means |
| `statistic` | the test statistic (z-value) |
| `pValue` | the significance (p-value) |
| `test` | name of test used |

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['age']
ex1 = replace(ex1, ex1=="89 OR OLDER", "90")
ts_z_is(df1['sex'], ex1)

#Example 2: vectors
scores = c(20,50,80,15,40,85,30,45,70,60, NA, 90,25,40,70,65, NA, 70,98,40)
groups = c("nat.","int.","int.","nat.","int.", "int.","nat.","nat.","int.",
"int.","int.","int.","int.","int.","nat.", "int." ,NA,"nat.","int.","int.")
ts_z_is(groups, scores)
```

---

## `ts_z_os` *One-Sample Z Test*

### Description

This test is often used if there is a large sample size. For smaller sample sizes, a Student t-test is usually used.

The assumption about the population (null hypothesis) for this test is a pre-defined mean, i.e. the (arithmetic) mean that is expected in the population. If the p-value (significance) is then below a pre-defined threhold (usually 0.05), the assumption is rejected.

This function is shown in this YouTube video and the test is also described at PeterStatistics.com

### Usage

```
ts_z_os(data, mu = NULL, sigma = NULL)
```

### Arguments

| | |
|---|---|
| `data` | A vector or dataframe with the data as numbers |
| `mu` | optional hypothesized mean, otherwise the midrange will be used |
| `sigma` | population standard deviation, if NULL sample results will be used |

## Details

The formula used is:

$$z = \frac{\bar{x} - \mu_{H_0}}{SE}$$

$$sig = 2 \times \left(1 - \Phi\left(\left|z\right|\right)\right)$$

With:

$$SE = \frac{\sigma}{\sqrt{n}}$$

$$\sigma \approx s = \sqrt{\frac{\sum_{i=1}^{n} \left(x_i - \bar{x}\right)^2}{n - 1}}$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

*Symbols used:*

- $\Phi\left(\ldots\right)$ the cumulative distribution function of the standard normal distribution
- $\bar{x}$ the sample mean
- $\mu_{H_0}$ the hypothesized mean in the population
- $SE$ the standard error (i.e. the standard deviation of the sampling distribution)
- $n$ the sample size (i.e. the number of scores)
- $s$ the unbiased sample standard deviation
- $x_i$ the i-th score

## Value

A dataframe with:

| | |
|---|---|
| `mu` | the hypothesized mean |
| `sample.mean` | the sample mean |
| `statistic` | the test statistic |
| `pValue` | the significance (p-value) |
| `testUsed` | name of test used |

## Before, After and Alternatives

Before this you might want to create a binned frequency table or a visualisation: `tab_frequency_bins`, to create a binned frequency table. `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

After this you might want an effect size measure: `es_cohen_d_os`, for for Cohen d'. `es_hedges_g_os`, for Hedges g. `es_common_language_os`, for the Common Language Effect Size.

Alternative Tests: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### Examples

```
#Example 1: dataframe
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
ex1 = df2['Gen_Age']
ts_z_os(ex1)
ts_z_os(ex1, mu=22, sigma=12.1)

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
ts_z_os(ex2)
```

---

## `ts_z_ps` *Z-test (Paired Samples)*

---

### Description

This test is often used if there is a large sample size. For smaller sample sizes, a Student t-test is usually used.

The assumption about the population (null hypothesis) for this test is a pre-defined difference between two means, usually zero (i.e. the difference between the (arithmetic) means is zero, they are the same in the population). If the p-value (significance) is then below a pre-defined threhold (usually 0.05), the assumption is rejected.

### Usage

```
ts_z_ps(field1, field2, dmu = 0, dsigma = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | the scores on the first variable |
| `field2` | the scores on the second variable |
| `dmu` | difference according to null hypothesis (default is 0) |
| `dsigma` | population standard deviation of the difference, if NULL sample results will be used |

### Details

The formula used is:

$$z_p = \frac{\bar{d} - d_{H0}}{SE}$$

$$sig. = 2 \times \left(1 - \Phi\left(\left|z_p\right|\right)\right)$$

With:

$$\bar{d} = \mu_1 - \mu_2 \approx \bar{x}_1 - \bar{x}_2$$

$$SE = \sqrt{\frac{\sigma_d^2}{n}} \approx \sqrt{\frac{\sigma_s^2}{n}}$$

$$s_d^2 = \frac{\sum_{i=1}^{n}\left(d_i - \bar{d}_i\right)^2}{n - 1}$$

$$d_i = x_{i,1} - x_{i,2}$$
$$\bar{d} = \frac{\sum_{i=1}^{n} d_i}{n}$$

**Symbols used:**

- $n$ the number of pairs (sample size)
- $x_{i,1}$ the i-th score of the first variable
- $x_{i,2}$ the i-th score of the second variable
- $d_{H0}$ the expected difference in the population
- $\Phi\left(\dots\right)$, cumulative density function of the standard normal distribution.

**Alternatives**

*library(DescTools)*

dfr = na.omit(data.frame(var1, var2))

ZTest(dfr$var1, dfr$var2, sd_pop=sqrt(var(dfr$var1-dfr$var2)), paired=TRUE)

## Value

A dataframe with:

| | |
|---|---|
| `n` | the number of scores |
| `z` | the test statistic (z-value) |
| `p-Value` | the significance (p-value) |

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

# `vi_bar_clustered` *Clustered / Multiple Bar Chart*

## Description

A bar-chart is defined as “a graph in which bars of varying height with spaces between them are used to display data for variables defined by qualities or categories” (Zedeck, 2014, p. 20).

The bars can be split into multiple bars based on another variable. This is then known as a multiple bar-chart (Kemp, 2004, p. 150) or clustered bar-chart (Brase, 2009, p. 50; Griffith, 2007, p. 168).

It can be defined as “a bar chart for comparing the frequencies of a categorical variable in two or more situations” (Upton & Cook, 2014, p. 283).

The first field will be placed on the horizontal axis, and the second used for the clusters.

## Usage

```
vi_bar_clustered(
  field1,
  field2,
  order1 = NULL,
  order2 = NULL,
  percent = c(NULL, "all", "row", "column")
)
```

### Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `order1` | : optional list with order for categories of field1 |
| `order2` | : optional list with order for categories of field2 |
| `percent` | : optional which percentages to show. Either "none" (default), "all", "row", "column" |

### Value

clustered bar chart

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Brase, C. (2009). *Understandable statistics* (9th ed.). Houghton MIfflin.

Griffith, A. (2007). *SPSS for dummies*. Wiley.

Kemp, S. M., & Kemp, S. (2004). *Business statistics demystified*. McGraw-Hill.

Upton, G., & Cook, I. (2014). *Oxford: Dictionary of statistics* (3rd ed.). Oxford University Press.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

### Examples

```
#Example 1: Clustered Bar Chart in percentages
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
vi_bar_clustered(df1[['mar1']], df1[['sex']], percent="column")

#Example 2: Specified order
orderR = c("DIVORCED", "WIDOWED", "SEPARATED", "MARRIED", "NEVER MARRIED")
orderC = c("MALE", "FEMALE")
vi_bar_clustered(df1[['mar1']], df1[['sex']], order1=orderR, order2=orderC)
```

---

## `vi_bar_dual_axis` *Dual-Axis Bar Chart*

### Description

A dual axis bar-chart is a bar-chart with two vertical axis. In this function it will show both the count and cumulative proportion.

This chart could be used with a single ordinal variable.

The visualisation is also described at PeterStatistics.com

## Usage

```
vi_bar_dual_axis(data, varname = NULL)
```

## Arguments

| | |
|---|---|
| `data` | the data from which to create a Pareto chart |
| `varname` | a name for the data, if not provided the name of the data variable is used |

## Value

a chart in the plot window

## Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`, for a frequency table

After visualisation you might want some descriptive measures: `me_consensus`, for the Consensus. `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_median`, for the Median. `me_quantiles`, for Quantiles. `me_quartiles`, for Quartiles / Hinges. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

Alternatives for this visualisation could be: `vi_bar_stacked_single`, or Single Stacked Bar-Chart.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
vi_bar_dual_axis(ex1);
vi_bar_dual_axis(ex1, varname="marital status");
```

---

# `vi_bar_simple` *Simple Bar-Chart*

---

## Description

A bar-chart is defined as “a graph in which bars of varying height with spaces between them are used to display data for variables defined by qualities or categories” (Zedeck, 2014, p. 20).

A YouTube video on pie charts.

This function is shown in this YouTube video and the visualisation is also described at PeterStatistics.com

## Usage

```
vi_bar_simple(data, varname = NULL, height = "count")
```

## Arguments

| | |
|---|---|
| `data` | A vector or dataframe |
| `varname` | Optional name for the variable |
| `height` | Optional to indicate what the height should represent |

## Details

The function uses the basic R's graphics library *barplot* function.

As a guideline for the size of the bar there is a rule of thumb known as the 'three quarter high rule' (Pitts, 1971). It means that the height of the vertical axis should be 3/4 of the length of the horizontal axis. So if the horizontal axis is 20 cm long, the vertical axis should be 3/4 * 20 = 15 cm high.

According to Singh (2009) vertical bars (instead of horizontal bars) are preferred since they are easier on the eye. However if you have long category names some names might become unreadable. A bar chart with the bars placed horizontally might then be preferred.

One of the earliest found bar-charts from William Playfair (1786) has the bars placed horizontally. There is an earlier bar chart by Oresme (1486), but that is used more for a theoretical concept, than for descriptive statistics.

## Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`

After visualisation you might want some descriptive measures: `me_mode`, for the mode. `me_qv`, for Measures of Qualitative Variation.

or perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

Alternatives for this visualisation could be: `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Oresme, N. (1486). *Tractatus de latitudinibus formarum*. (B. Pelacani da Parma, Ed.). Mathaeus Cerdonis.

Pitts, C. E. (1971). *Introduction to educational psychology: An operant conditioning approach*. Crowell.

Playfair, W. (1786). *The commercial and political atlas*. Debrett; Robinson; and Sewell.

Singh, G. (2009). *Map work and practical geography* (4th ed). Vikas Publishing House Pvt Ltd.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
vi_bar_simple(ex1);
vi_bar_simple(ex1, varname="marital status", height="percent");

#Example 2: a list
ex2 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
vi_bar_simple(ex2);
```

---

# `vi_bar_stacked_multiple`

*Multiple Stacked Bar-Chart*

---

## Description

To visualise an ordinal variable, it often makes sense to stack the results. Stacking the results creates a compound bar chart, or sometimes stacked bar chart (Wilkinson, 2005, p. 157) or component bar chart (Zedeck, 2014, p. 54). It can be defined as: “a bar chart showing multiple bars stacked at each x-axis category, each representing a value of the stacking variable” (Upton & Cook, 2014, p. 88).

Instead of one bar (see **vi_bar_stacked_single()**), we can create two or more (one for each group). This could then be considered a multiple compound bar-chart.

## Usage

```
vi_bar_stacked_multiple(catField, ordField, levels = NULL, ...)
```

## Arguments

| | |
|---|---|
| `catField` | list or dataframe with the categories |
| `ordField` | list or dataframe with the scores |
| `levels` | optional list with the scores in order |
| `...` | optional, other parameters for use in barplot function |

## Details

This function is more like a wrapper for the **barplot()** from R *graphics* library.

## Value

multiple stacked bar-chart

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Upton, G., & Cook, I. (2014). *Oxford: Dictionary of statistics* (3rd ed.). Oxford University Press.

Wilkinson, L. (2005). *The grammar of graphics* (2nd ed). Springer.

### Examples

```
file1 = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(file1, sep=",", na.strings=c("", "NA"))
vi_bar_stacked_multiple(df1[['mar1']], df1[['accntsci']], ylab= "percent", col=1:5)

cats = c(1, 1, 2, 2, 2, 3, 3, 3, 3)
scor = c(1, 2, 1, 1, 2, 1, 1, 1, 2)
vi_bar_stacked_multiple(cats, scor, ylab= "percent", col=1:5)
```

---

## `vi_bar_stacked_single` *Single Stacked Bar-Chart*

---

### Description

A regular bar-chart but with the bars on top of each other, instead of next to each other. This is called a compound bar chart, stacked bar chart (Wilkinson, 2005, p. 157) or component bar chart (Zedeck, 2014, p. 54).

It can be defined as: “a bar chart showing multiple bars stacked at each x-axis category, each representing a value of the stacking variable” (Upton & Cook, 2014, p. 88).

The visualisation is also described at PeterStatistics.com

### Usage

```
vi_bar_stacked_single(data, catCoding = NULL, orientation = c("h", "v"))
```

### Arguments

| | |
|---|---|
| `data` | the data from which to create the bar-chart |
| `catCoding` | optional vector with the order for the bars |
| `orientation` | optional to indicate horizontal or vertical chart Either `"h"` (default) or `"v"` |

### Details

This function basically uses barplot(...,beside = FALSE) from R’s *graphics* library

### Value

The chart.

### Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`, for a frequency table

After visualisation you might want some descriptive measures: `me_consensus`, for the Consensus. `me_hodges_lehmann_os`, for the Hodges-Lehmann Estimate (One-Sample). `me_median`, for the Median. `me_quantiles`, for Quantiles. `me_quartiles`, for Quartiles / Hinges. `me_quartile_range`, for Interquartile Range, Semi-Interquartile Range and Mid-Quartile Range.

or perform a test: `ts_sign_os`, for One-Sample Sign Test. `ts_trinomial_os`, for One-Sample Trinomial Test. `ts_wilcoxon_os`, for One-Sample Wilcoxon Signed Rank Test.

Alternatives for this visualisation could be: `vi_bar_dual_axis`, for Dual-Axis Bar Chart.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Upton, G. J. G., & Cook, I. (2014). *Dictionary of statistics* (3rd ed.). Oxford University Press.

Wilkinson, L. (2005). *The grammar of graphics* (2nd ed). Springer.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: Text dataframe
ex1 = df2[['Teach_Motivate']]
order = c("Fully Disagree", "Disagree", "Neither disagree nor agree", "Agree", "Fully agree")
vi_bar_stacked_single(ex1, catCoding=order)
vi_bar_stacked_single(ex1, catCoding=order, orientation="v");

#Example 2: Numeric data
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
vi_bar_stacked_single(ex2);
```

---

## `vi_boxplot_single` *Box (and Whisker) Plot*

### Description

A box plot is a little more complex visualisation than a histogram. It shows the five quartiles (e.g. minimum, 1st quartile, median, 3rd quartile, and maximum). It can also be adjusted to show so-called outliers.

This function is shown in this YouTube video and the visualisation is described at PeterStatistics.com

### Usage

```
vi_boxplot_single(data, varname = NULL)
```

## Arguments

|  |  |
|---|---|
| `data` | list or dataframe |
| `varname` | optional name to display on vertical axis |

## Details

This was actually a 'range chart' (Spear, 1952, p. 166) but somehow it is these days referred to as a box-and-whisker plot as named by Tukey (1977, p. 39)

The function uses the **boxplot()** function from the *graphics* library. If you want to modify more things you might want to use that function.

## Value

boxplot

## Before, After and Alternatives

Before this you might want to create a binned frequency table `tab_frequency_bins`, to create a binned frequency table.

After this you might want some descriptive measures: `me_mode_bin`, for Mode for Binned Data. `me_mean`, for different types of mean. `me_variation`, for different Measures of Quantitative Variation.

Or a perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

Alternative Visualisations: `vi_histogram`, for a Histogram. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Spear, M. E. (1952). *Charting statistics*. McGraw-Hill.

Tukey, J. W. (1977). *Exploratory data analysis*. Addison-Wesley Pub. Co.

## Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: dataframe
ex1 = df2['Gen_Age']
vi_boxplot_single(ex1);

#Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
vi_boxplot_single(ex2);
```

---

## `vi_boxplot_split` *Split Box Plot*

---

### Description

Based on a categorical field the scores for each category are plotted in a separate boxplot and each of them is placed underneath each other.

See **vi_boxplot_single()** for more details on boxplots.

### Usage

```
vi_boxplot_split(catField, scaleField, categories = NULL, ...)
```

### Arguments

| | |
|---|---|
| `catField` | list or dataframe with the categories |
| `scaleField` | list or dataframe with the scores |
| `categories` | optional list with categories to use |
| `...` | other parameters for use in boxplot function |

### Value

The split boxplot

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

## `vi_butterfly_chart` *Butterfly Chart / Tornado Chart / Pyramid Chart*

---

### Description

A special case of diverging bar charts when only comparing two categories.

Depending on the ordering of the results different names exist. I've chosen to use 'butterfly' if no ordering is done, 'pyramid' if they are ordered from small to large, and 'tornado' when going from large to small.

### Usage

```
vi_butterfly_chart(
  field1,
  field2,
  categories1 = NULL,
  categories2 = NULL,
  variation = "butterfly",
  roundHigh = 5
)
```

## Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with selection of categories of field1 |
| `categories2` | : optional list with selection of categories of field2 |
| `variation` | : optional order of the bars. Either "butterfly" (default), "tornado", or "pyramid" |
| `roundHigh` | : optional to adjust number of tickmarks on horizontal axis |

## Details

The term *butterfly chart* can for example be found in Hwang and Yoon (2021, p. 25).

The term *tornado diagrom* can be found in the guide from the Project Management Institute (2013, p. 338). The term *funnel chart* is also sometimes used (for example Jamsa (2020, p. 135)), but this is also a term sometimes used for a more analytical scatterplot used for some specific analysis.

The term *pyramid chart* can for example be found in Schwabish (2021, p. 185). It is very often used for comparing age distributions.

## Value

plot

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Hwang, J., & Yoon, Y. (2021). Data analytics and visualization in quality analysis using Tableau. CRC Press.

Jamsa, K. (2020). Introduction to data mining and analytics: With machine learning in R and Python. Jones & Bartlett Learning.

Project Management Institute (Ed.). (2013). A guide to the project management body of knowledge (5th ed.). Project Management Institute, Inc.

Schwabish, J. (2021). Better data visualizations: A guide for scholars, researchers, and wonks. Columbia University Press.

## Examples

```
#Example
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
vi_butterfly_chart(df1[['mar1']], df1[['sex']], roundHigh=100)
vi_butterfly_chart(df1[['mar1']], df1[['sex']], variation="tornado", roundHigh=100)
vi_butterfly_chart(df1[['mar1']], df1[['sex']], variation="pyramid", roundHigh=100)
```

---

## `vi_cleveland_dot_plot` *Cleveland Dot Plot*

---

### Description

A Cleveland dot plot (Cleveland & McGill, 1987) is a bar chart where instead of bars a dot is placed at the center of the top of the bar (and then the bars removed). It is a dot plot only showing the top dot.This requires less ink.

The function simply uses the dotplot() function from the lattice library.

A video on (Cleveland) dot plots is available here.

This function is shown in this YouTube video and the visualisation is also described at PeterStatistics.com

### Usage

```
vi_cleveland_dot_plot(data, size = 2)
```

### Arguments

| | |
|---|---|
| `data` | the data from which to create the plot |
| `size` | the size of the dots (default is 2) |

### Value

chart the Cleveland dot plot

### Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`

After visualisation you might want some descriptive measures: `me_mode`, for the mode. `me_qv`, for Measures of Qualitative Variation.

or perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

Alternatives for this visualisation could be: `vi_bar_simple`, for Simple Bar Chart. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Cleveland, W. S., & McGill, R. (1984). Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association, 79*(387), 531–554. https://doi.org/10.2307/2288400

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
vi_cleveland_dot_plot(ex1);

#Example 2: a list
ex2 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
vi_cleveland_dot_plot(ex2);
```

---

# `vi_dot_plot` *Dot Plot*

## Description

The Oxford Dictionary of Statistics defines a dot plot as "an alternative to a bar chart or line graph when there are very few data values. Each value is recorded as a dot, so that the frequencies for each value can easily be counted" (Upton & Cook, 2014, p. 129).

This function uses ggplot2 geom_dotplot() to create a simple dot plot.

A YouTube video on dot plots.

This function is shown in this YouTube video and the visualisation is also described at PeterStatistics.com

## Usage

```
vi_dot_plot(data, stackRatio = 1, dotSize = 1)
```

## Arguments

| | |
|---|---|
| `data` | the data from which to create the dot plot |
| `stackRatio` | ratio on how close the dots are to each other |
| `dotSize` | indicator for how big the dots need to be |

## Details

In the definition a *bar chart* is mentioned. A bar chart can be defined as “a graph in which bars of varying height with spaces between them are used to display data for variables defined by qualities or categories” (Zedeck, 2014, p. 20). Together this indicates that a dot plot is used for categorical data.

However, Zedeck sees the dot plot as an alternative name for a scatterplot, which is for continuous data. A third version comes from the Cambridge Dictionary of Statistics: "A more effective display than a number of other methods, for example, pie charts and bar charts, for displaying quantitative data which are labelled" (Everitt, 2004, p. 123). They also show an example where we see a categorical variable on one axis, and a continuous variable on another.

This function was only for the original definition for categorical data.

## Value

chart the dot plot

## Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`

After visualisation you might want some descriptive measures: `me_mode`, for the mode. `me_qv`, for Measures of Qualitative Variation.

or perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

Alternatives for this visualisation could be: `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_pareto_chart`, for Pareto Chart. `vi_pie`, for Pie Chart.

## Author(s)

P. Stikker. Companion Website, YouTube Channel

## References

Everitt, B. (2004). *The Cambridge dictionary of statistics* (2nd ed). Cambridge University Press.

Upton, G. J. G., & Cook, I. (2014). *Dictionary of statistics* (3rd ed.). Oxford University Press.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

## Examples

```
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")
vi_dot_plot(data, stackRatio=1.5, dotSize=2)
```

---

# `vi_histogram` *Histogram*

## Description

A histogram is a bit like a bar chart for a scale variable. You would create some bins, and then plot these as bars.

This function is shown in this YouTube video and the visualisation is described at PeterStatistics.com

## Usage

```
vi_histogram(data, xlbl = NULL, ylbl = NULL, ...)
```

## Arguments

| | |
|---|---|
| `data` | list or dataframe |
| `xlbl` | optional label for the horizontal axis |
| `ylbl` | optional label for the vertical axis |
| `...` | other parameters for use in hist function |

## Details

This function is just using some defaults for the **hist()** function from R's *graphics* library.

To set the bins, the *breaks* argument can be used. This could be a pre-set number based on a calculation, a specific rule (e.g. bins="sturges"), or a list with the cut-off points.

If your bins are of equal width, a true histogram than actually should show frequency densities (Pearson, 1895, p. 399). These are the frequencies divided by the bin-width. This can be done using *freq=FALSE* parameter.

## Value

The histogram

## Before, After and Alternatives

Before this you might want to create a binned frequency table `tab_frequency_bins`, to create a binned frequency table.

After this you might want some descriptive measures: `me_mode_bin`, for Mode for Binned Data. `me_mean`, for different types of mean. `me_variation`, for different Measures of Quantitative Variation.

Or a perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

Alternative Visualisations: `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_stem_and_leaf`, for a Stem-and-Leaf Display.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Pearson, K. (1895). Contributions to the mathematical theory of evolution. II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London. (A.)*, 186, 343–414. doi:10.1098/rsta.1895.0010

## Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
df2 = read.csv(file2, sep=';', na.strings=c("", "NA"))
#Example 1: dataframe
ex1 = df2['Gen_Age']
vi_histogram(ex1);
vi_histogram(ex1, freq=FALSE);

#Example 2: Numeric list
```

```
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
vi_histogram(ex2);
```

---

## `vi_histogram_split` *Split Histogram*

### Description

Based on a categorical field the scores for each category are plotted in a separate histogram and each of the histograms is placed underneath each other.

See **vi_histogram()** for more details on histograms.

### Usage

```
vi_histogram_split(catField, scaleField, categories = NULL, ...)
```

### Arguments

| | |
|---|---|
| `catField` | list or dataframe with the categories |
| `scaleField` | list or dataframe with the scores |
| `categories` | optional list with categories to use |
| `...` | other parameters for use in geom_histogram function |

### Value

The split histogram

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

---

## `vi_pareto_chart` *Pareto Chart*

### Description

The Pareto Chart gets its name from the Pareto Principle, which is named after Vilfredo Pareto. This principle states that roughly 80% of consequencies come from 20% of causes (Pareto, 1896).

Unfortunately, there is no general agreed upon definition of a Pareto diagram. The most general description I've found was by Kemp and Kemp (2004) who mention it is a name for a bar chart if the order of the bars have no meaning (i.e. for a nominal variable), and they only mention that often the bars are then placed in decreasing order. According to some authors a Pareto diagram is any diagram with the bars in order of size (Joiner, 1995; WhatIs.com, n.d.), while others suggest that a line representing the cumulative relative frequencies should also be included (Weisstein, 2002). Upton and Cook (2014) also add that the bars should not have any gaps, but many other authors ignore this.

The following definition by the author is used: a bar chart where the bars are placed in descending order of frequency. Usually an ogive is added in the chart as well.

An ogive (oh-jive) is: "the graphs of cumulative frequencies" (Kenney, 1939).

A video on Pareto charts is available here.

This function is shown in this YouTube video and the visualisation is also described at PeterStatistics.com

## Usage

```
vi_pareto_chart(data, varname = NULL)
```

## Arguments

| | |
|---|---|
| `data` | the data from which to create a Pareto chart |
| `varname` | a name for the data, if not provided the name of the data variable is used |

## Value

a Pareto chart in the plot window

## Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`

After visualisation you might want some descriptive measures: `me_mode`, for the mode. `me_qv`, for Measures of Qualitative Variation.

or perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

Alternatives for this visualisation could be: `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pie`, for Pie Chart.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Joiner. (1995). Pareto charts: Plain & simple. Joiner Associates.

Kemp, S. M., & Kemp, S. (2004). *Business statistics demystified*. McGraw-Hill.

Kenney, J. F. (1939). *Mathematics of statistics; Part one*. Chapman & Hall.

Pareto, V. (1896). *Cours d'économie politique* (Vol. 1). Lausanne.

Upton, G. J. G., & Cook, I. (2014). *Dictionary of statistics* (3rd ed.). Oxford University Press.

Weisstein, E. W. (2002). *CRC concise encyclopedia of mathematics* (2nd ed.). Chapman & Hall/CRC.

WhatIs.com. (n.d.). What is Pareto chart (Pareto distribution diagram)? - Definition from WhatIs.com. Retrieved April 20, 2014, from http://whatis.techtarget.com/definition/Pareto-chart-Pareto-distribution-diagram

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
vi_pareto_chart(ex1);

#Example 2: a list
ex2 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
vi_pareto_chart(ex2);
```

---

# `vi_pie` *Pie Chart*

## Description

A pie-chart is a “graphic display in which a circle is cut into wedges with the area of each wedge being proportional to the percentage of cases in the category represented by that wedge” (Zedeck, 2014, p. 260).

A video on pie charts is available here.

This function is shown in this YouTube video and the visualisation is also described at PeterStatistics.com

## Usage

```
vi_pie(data, labels = c("count", "percent", "both", "none"))
```

## Arguments

| | |
|---|---|
| `data` | the data for which to create a pie-chart from |
| `labels` | what to show besides the labels Either "count" (default), "percent", "none", or "both" |

## Details

It is possible to either show only the labels (label="none"), the counts (label="counts"), the percentages (label="percent"), or both count and percent (label="both").

The function uses the basic R’s graphics library *pie* function, rotated and counter clockwise.

The pie-chart is quite popular and often used, but actually has a few disadvantages. It can only show relative frequencies. To show other frequencies the numbers themselves have to be added. A circle has 360 degrees, equal to 100%. So by multiplying the relative frequencies with 360, the degrees for each category can be found. This means that visually the pie-chart can only show the relative frequencies.

Another disadvantage is when the relative frequencies are close to each other, the differences are not easily seen in a circle diagram.

As a third disadvantage, when there are many categories the circle diagram will look very busy and not easily to read.

People also have more difficulty with comparing areas and angles (what you do when looking at a pie-chart) than comparing heights (what is done with a bar-chart).

Also often a 3D effect is added, but this actually makes comparisons of the slices even more difficult.

he earliest found circle diagram is found on the inlay of a book by William Playfair (1801). The name 'pie chart' might come from a misspelling of the word Pi. Pi is often associated with a circle. It might also simply come from the resemblances with a pie (as in apple-pie). However Srivastava and Rego (2011) put forward another belief that it is named after a royal French cook Pie, who served dishes in a pie-chart shape.

## Value

chart the pie chart

## Before, After and Alternatives

Before the visualisation you might first want to get an impression using a frequency table: `tab_frequency`

After visualisation you might want some descriptive measures: `me_mode`, for the mode. `me_qv`, for Measures of Qualitative Variation.

or perform a test: `ts_pearson_gof`, for Pearson Chi-Square Goodness-of-Fit Test. `ts_freeman_tukey_gof`, for Freeman-Tukey Test of Goodness-of-Fit. `ts_freeman_tukey_read`, for Freeman-Tukey-Read Test of Goodness-of-Fit. `ts_g_gof`, for G (Likelihood Ratio) Goodness-of-Fit Test. `ts_mod_log_likelihood_gof`, for Mod-Log Likelihood Test of Goodness-of-Fit. `ts_multinomial_gof`, for Multinomial Goodness-of-Fit Test. `ts_neyman_gof`, for Neyman Test of Goodness-of-Fit. `ts_powerdivergence_gof`, for Power Divergence GoF Test.

Alternatives for this visualisation could be: `vi_bar_simple`, for Simple Bar Chart. `vi_cleveland_dot_plot`, for Cleveland Dot Plot. `vi_dot_plot`, for Dot Plot. `vi_pareto_chart`, for Pareto Chart.

## Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

## References

Playfair, W. (1801). *The statistical breviary: Shewing the resources of every state and kingdom*. T. Bensley. http://archive.org/details/statisticalbrev00playgoog

Srivastava, T. N., & Rego, S. (2011). *Business research methodology*. Tata McGraw-Hill.

Zedeck, S. (Ed.). (2014). *APA dictionary of statistics and research methods*. American Psychological Association.

## Examples

```
#Example 1: dataframe
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
ex1 = df1['mar1']
vi_pie(ex1);
vi_pie(ex1, labels="percent");
vi_pie(ex1, labels="none");
vi_pie(ex1, labels="both");

#Example 2: a list
ex2 = c("MARRIED", "DIVORCED", "MARRIED", "SEPARATED", "DIVORCED", "NEVER MARRIED",
```

```
"DIVORCED", "DIVORCED", "NEVER MARRIED", "MARRIED", "MARRIED", "MARRIED", "SEPARATED",
"DIVORCED", "NEVER MARRIED", "NEVER MARRIED", "DIVORCED", "DIVORCED", "MARRIED")
vi_pie(ex2);
```

---

## `vi_spine_plot` *Spine Plot*

### Description

A spine plot is similar to a multiple stacked bar-chart, but "the difference is that the bars fill the plot vertically so the shading gives us proportions instead of counts. Also, the width of each bar varies, reflecting the marginal proportion of observations in each workshop" (Muenchen, 2006, p. 286)

It is a chart you could use when with two nominal variables and do not have a clear independent and dependent variable. Otherwise a multiple/clustered bar-chart might be preferred.

### Usage

```
vi_spine_plot(field1, field2, categories1 = NULL, categories2 = NULL)
```

### Arguments

| | |
|---|---|
| `field1` | : dataframe field with categories for the rows |
| `field2` | : dataframe field with categories for the columns |
| `categories1` | : optional list with selection of categories of field1 |
| `categories2` | : optional list with selection of categories of field2 |

### Details

The naming of this diagram is unfortunately not very clear. I use the term 'spine plot' as a special case of a Mosaic Plot. Mosaic Plots are often attributed to Hartigan and Kleiner (for example by Friendly (2002, p. 90)). Earlier versions are actually known, for example Walker (1874, p. PI XX). Hartigan and Kleiner (1981) start their paper with a Mosaic Plot for a cross table, but end it with showing Mosaic Plots for multiple dimension cross tables.

A Marimekko Chart is simply an alternative name for the Mosaic Plot, although according to Wikipedia "mosaic plots can be colored and shaded according to deviations from independence, whereas Marimekko charts are colored according to the category levels" (Wikipedia, 2022).

The term 'Spine Plot' itself is often attributed to Hummel, but I've been unable to hunt down his original article: Linked bar charts: Analysing categorical data graphically. Computational Statistics 11: 23–33.

### Value

spine plot

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Carvalho, T. (2021, April 10). Marimekko Charts with Python’s Matplotlib. Medium. https://towardsdatascience.com/ma charts-with-pythons-matplotlib-6b9784ae73a1

Friendly, M. (2002). A brief history of the mosaic display. *Journal of Computational and Graphical Statistics, 11*(1), 89–107. https://doi.org/10.1198/106186002317375631

Hartigan, J. A., & Kleiner, B. (1981). Mosaics for contingency tables. In W. F. Eddy (Ed.), Proceedings of the 13th Symposium on the Interface (pp. 268–273). Springer. https://doi.org/10.1007/978-1-4613-9464-8_37

Muenchen, R. A. (2009). *R for SAS and SPSS Users*. Springer.

Walker, F. A. (1874). *Statistical atlas of the United States based on the results of the ninth census 1870*. Census Office.

Wikipedia. (2022). Mosaic plot. In Wikipedia. https://en.wikipedia.org/w/index.php?title=Mosaic_plot&oldid=1089465

### Examples

```
#Example
dataFile = "https://peterstatistics.com/Packages/ExampleData/GSS2012a.csv"
df1 <- read.csv(dataFile, sep=",", na.strings=c("", "NA"))
vi_bar_clustered(df1[['mar1']], df1[['sex']], percent="column")
```

---

## `vi_stem_and_leaf` *Stem-and-Leaf Display*

### Description

A stem-and-leaf display is defined as: "a method of displaying data in which each observation is split into two parts labelled the ‘stem’ and the ‘leaf’" (Everitt, 2004, p. 362). A diagram that could be used to visualize scale variables, created by Tukey (1972, p. 296).

In some variations of this, the cumulative frequencies are also shown, but currently this function does not provide for that.

This function is shown in this YouTube video and the visualisation is described at PeterStatistics.com

### Usage

```
vi_stem_and_leaf(data, key = NULL)
```

### Arguments

| | |
|---|---|
| `data` | list with the numeric data |
| `key` | optional factor to use for the stems |

### Value

prints the display in console and returns a dataframe with the stems and leafs

### Before, After and Alternatives

Before this you might want to create a binned frequency table `tab_frequency_bins`, to create a binned frequency table.

After this you might want some descriptive measures: `me_mode_bin`, for Mode for Binned Data. `me_mean`, for different types of mean. `me_variation`, for different Measures of Quantitative Variation.

Or a perform a test: `ts_student_t_os`, for One-Sample Student t-Test. `ts_trimmed_mean_os`, for One-Sample Trimmed (Yuen or Yuen-Welch) Mean Test. `ts_z_os`, for One-Sample Z Test.

Alternative Visualisations: `vi_boxplot_single`, for a Box (and Whisker) Plot. `vi_histogram`, for a Histogram.

### Author(s)

P. Stikker. Companion Website, YouTube Channel, Patreon donations

### References

Everitt, B. (2004). *The Cambridge dictionary of statistics* (2nd ed.). Cambridge University Press.

Tukey, J. W. (1972). Some graphic and semigraphic displays. In T. A. Bancroft & S. A. Brown (Eds.), *Statistical Papers in Honor of George W. Snedecor* (pp. 293–316). Iowa State University Press.

### Examples

```
file2 = 'https://peterstatistics.com/Packages/ExampleData/StudentStatistics.csv'
studentDf = read.csv(file2, sep=';', na.strings=c("", "NA"))
# Example 1: dataframe
ex1 = studentDf[['Gen_Age']]
vi_stem_and_leaf(ex1);

# Example 2: Numeric list
ex2 = c(1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5)
vi_stem_and_leaf(ex2);
```

# Index