Package ‘vcmeta’

June 17, 2022

Type    Package
Title   Varying Coefficient Meta-Analysis
Version 1.1.0

Description  Implements functions for varying coefficient meta-analysis methods. These methods do not assume effect size homogeneity. Subgroup effect size comparisons, general linear effect size contrasts, and linear models of effect sizes based on varying coefficient methods can be used to describe effect size heterogeneity. Varying coefficient meta-analysis methods do not require the unrealistic assumptions of the traditional fixed-effect and random-effects meta-analysis methods.
For details see: Statistical Methods for Psychologists, Volume 5, <https://dgbonett.sites.ucsc.edu/>.

URL    https://github.com/dgbonett/vcmeta

BugReports  https://github.com/dgbonett/vcmeta/issues
License GPL-3

Encoding UTF-8

Imports stats, mathjaxr, Rdpack, ggplot2
RdMacros Rdpack, mathjaxr
Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author  Douglas G. Bonett [aut, cre],
        Robert J. Calin-Jageman [ctb]
Maintainer Douglas G. Bonett <dgbonett@ucsc.edu>

Repository CRAN

Date/Publication 2022-06-17 07:00:02 UTC
### Topics Documented:

- `ci.fisher` ................................................. 4
- `cor.from.t` ............................................... 4
- `meta.ave.agree` .......................................... 5
- `meta.ave.cor` ............................................. 6
- `meta.ave.cronbach` ...................................... 7
- `meta.ave.gen` ............................................. 8
- `meta.ave.gen.cc` ......................................... 10
- `meta.ave.gen.rc` ......................................... 11
- `meta.ave.mean.ps` ......................................... 12
- `meta.ave.mean2` .......................................... 14
- `meta.ave.meanratio.ps` ................................... 15
- `meta.ave.meanratio2` ..................................... 16
- `meta.ave.odds` ........................................... 18
- `meta.ave.path` ............................................ 19
- `meta.ave.pbcor` ........................................... 20
- `meta.ave.plot` ............................................ 22
- `meta.ave.prop.ps` .......................................... 23
- `meta.ave.prop2` ........................................... 24
- `meta.ave.propratio2` .................................... 26
- `meta.ave.semipart` ....................................... 27
- `meta.ave.slope` ........................................... 28
- `meta.ave.spear` ........................................... 29
- `meta.ave.stdmean.ps` ..................................... 30
- `meta.ave.stdmean2` ....................................... 32
- `meta.ave.var` ............................................. 33
- `meta.chitest` ............................................. 34
- `meta.lc.agree` ............................................ 35
- `meta.lc.gen` ............................................... 36
- `meta.lc.mean.ps` .......................................... 37
- `meta.lc.mean1` ............................................ 38
- `meta.lc.mean2` ............................................ 39
- `meta.lc.meanratio.ps` .................................... 41
- `meta.lc.meanratio2` ...................................... 42
- `meta.lc.odds` ............................................. 43
- `meta.lc.prop.ps` .......................................... 45
- `meta.lc.prop1` ............................................ 46
- `meta.lc.prop2` ............................................ 47
- `meta.lc.propratio2` ...................................... 48
- `meta.lc.stdmean.ps` ....................................... 49
- `meta.lc.stdmean2` ........................................ 51
- `meta.lm.agree` ............................................ 52
- `meta.lm.cor` .............................................. 53
- `meta.lm.cor.gen` .......................................... 54
- `meta.lm.cronbach` ........................................ 55
- `meta.lm.gen` .............................................. 57
- `meta.lm.mean.ps` .......................................... 58
topics documented:

- meta.lm.mean1
- meta.lm.mean2
- meta.lm.meanratio.ps
- meta.lm.meanratio2
- meta.lm.odds
- meta.lm.prop.ps
- meta.lm.prop1
- meta.lm.prop2
- meta.lm.propratio2
- meta.lm.semipart
- meta.lm.spear
- meta.lm.stdmean.ps
- meta.lm.stdmean2
- meta.sub.cor
- meta.sub.cronbach
- meta.sub.pbcor
- meta.sub.semipart
- meta.sub.spear
- replicate.cor
- replicate.gen
- replicate.mean.ps
- replicate.mean2
- replicate.oddsratio
- replicate.plot
- replicate.prop2
- replicate.slope
- replicate.stdmean.ps
- replicate.stdmean2
- se.cor
- se.mean.ps
- se.mean2
- se.meanratio.ps
- se.meanratio2
- se.odds
- se.pbcor
- se.prop.ps
- se.prop2
- se.semipartial
- se.slope
- se.spear
- se.stdmean.ps
- se.stdmean2

Index
ci.fisher  

Fisher confidence interval for any type of correlation.

Description

This function computes a confidence interval for any type correlation using an estimated correlation and its standard error. This function should be used with the meta.ave.gen function when the effect size is a correlation. Use the estimated average correlation and its standard error from meta.ave.gen (when the effect size is a correlation) in the ci.fisher function to obtain a more accurate confidence interval for the population average correlation.

Usage

    ci.fisher(alpha, cor, se)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha</td>
<td>alpha value for 1-alpha confidence</td>
</tr>
<tr>
<td>cor</td>
<td>estimate of correlation</td>
</tr>
<tr>
<td>se</td>
<td>standard error of estimated correlation</td>
</tr>
</tbody>
</table>

Value

A 2-element vector with lower and upper bounds of the confidence interval

Examples

    ci.fisher(0.05, 0.50, .10)

    # Should return:
    # [1] 0.2802723 0.6699402

---

cor.from.t  

Compute Pearson correlation between paired measurements from t statistic

Description

This function computes the Pearson correlation between paired measurements using a reported paired-samples t statistic and other sample information. This correlation estimate is needed in several functions that analyze mean differences and standardized mean differences in paired-samples studies.
Usage

cor.from.t(m1, m2, sd1, sd2, t, n)

Arguments

m1  estimated mean for measurement 1
m2  estimated mean for measurement 2
sd1 estimated standard deviation for measurement 1
sd2 estimated standard deviation for measurement 2
t   value for paired-samples t-test
n   sample size

Value

Returns the sample Pearson correlation between the two paired measurements

Examples

cor.from.t(9.4, 9.8, 1.26, 1.40, 2.27, 30)

# Should return:
# [1] 0.7415209

meta.ave.agree  Confidence interval for an average G-index agreement coefficient

Description

Computes the estimate, standard error, and confidence interval for an G-index of agreement from two or more studies. This function assumes that two raters each provide a dichotomous rating to a sample of objects. The G-index of agreement is usually preferred to Cohen’s kappa.

Usage

meta.ave.agree(alpha, f11, f12, f21, f22, bystudy = TRUE)

Arguments

alpha  alpha level for 1-alpha confidence
f11    vector of frequencies in cell 1,1
f12    vector of frequencies in cell 1,2
f21    vector of frequencies in cell 2,1
f22    vector of frequencies in cell 2,2
bystudy logical to also return each study estimate (TRUE) or not
Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

Examples

```r
f11 <- c(17, 28, 19)
f12 <- c(43, 56, 49)
f21 <- c(3, 5, 5)
f22 <- c(37, 54, 39)
meta.ave.agree(.05, f11, f12, f21, f22, bystudy = TRUE)
# Should return:
# Estimate SE LL UL
# Average 0.08657934 0.05312584 -0.01754538 0.1907041
# Study 1 0.07692308 0.09776752 -0.11469775 0.2685439
# Study 2 0.14285714 0.08163265 -0.01713992 0.3028542
# Study 3 0.03448276 0.09279245 -0.14738711 0.2163526
```

Description

Computes the estimate, standard error, and confidence interval for an average Pearson or partial correlation from two or more studies. The sample correlations must be all Pearson correlations or all partial correlations. Use the meta.ave.gen function to meta-analyze any combination of Pearson, partial, or Spearman correlations.

Usage

```r
meta.ave.cor(alpha, n, cor, s, bystudy = TRUE)
```

Arguments

- alpha: alpha level for 1-alpha confidence
- n: vector of sample sizes
- cor: vector of estimated correlations
- s: number of control variables
- bystudy: logical to also return each study estimate (TRUE) or not
Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

```r
n <- c(55, 190, 65, 35)
cor <- c(.40, .65, .60, .45)
meta.ave.cor(.05, n, cor, 0, bystudy = TRUE)
```

# Should return:
# Estimate  SE  LL   UL
# Average  0.525 0.05113361 0.4176678 0.6178816
# Study 1  0.400 0.11430952 0.1506943 0.6014699
# Study 2  0.650 0.04200694 0.5594086 0.7252465
# Study 3  0.600 0.08000000 0.4171458 0.7361686
# Study 4  0.450 0.13677012 0.1373507 0.6811071
```

meta.ave.cronbach

Confidence interval for an average Cronbach alpha reliability

Description

Computes the estimate, standard error, and confidence interval for an average Cronbach reliability coefficient from two or more studies.

Usage

```r
meta.ave.cronbach(alpha, n, rel, r, bystudy = TRUE)
```
Arguments

alpha  alpha level for 1-alpha confidence
n      vector of sample sizes
rel    vector of sample reliabilities
r      number of measurements (e.g., items) used to compute each reliability
bystudy logical to also return each study estimate (TRUE) or not

Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

n <- c(583, 470, 546, 680)
rel <- c(.91, .89, .90, .89)
meta.ave.cronbach(.05, n, rel, 10, bystudy = TRUE)

# Should return:
# Estimate SE LL UL
# Average 0.8975 0.003256081 0.8911102 0.9038592
# Study 1 0.9100 0.005566064 0.8985763 0.9204108
# Study 2 0.8900 0.007579900 0.8743616 0.9041013
# Study 3 0.9000 0.006391375 0.8868623 0.9119356
# Study 4 0.8900 0.006297549 0.8771189 0.9018203

Description

Computes the estimate, standard error, and confidence interval for an average of any type of parameter from two or more studies.
**Usage**

```
meta.ave.gen(alpha, est, se, bystudy = TRUE)
```

**Arguments**

- `alpha` : alpha level for 1-alpha confidence
- `est` : vector of parameter estimates
- `se` : vector of standard errors
- `bystudy` : logical to also return each study estimate (TRUE) or not

**Value**

Returns a matrix. The first row is the average estimate across all studies. If `bystudy` is true, there is 1 additional row for each study. The matrix has the following columns:

- **Estimate** - the estimated effect size
- **SE** - standard error
- **LL** - lower limit of the confidence interval
- **UL** - upper limit of the confidence interval

**Examples**

```r
est <- c(.022, .751, .421, .287, .052, .146, .562, .904)
se <- c(.124, .464, .102, .592, .864, .241, .252, .318)
meta.ave.gen(.05, est, se, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th>Study</th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>0.393125</td>
<td>0.1561622</td>
<td>0.08705266</td>
<td>0.6991973</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.022000</td>
<td>0.1240000</td>
<td>-0.22103553</td>
<td>0.2650355</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.751000</td>
<td>0.4640000</td>
<td>-0.15842329</td>
<td>1.6604233</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.421000</td>
<td>0.1020000</td>
<td>0.22108367</td>
<td>0.6209163</td>
</tr>
<tr>
<td>Study 4</td>
<td>0.287000</td>
<td>0.5920000</td>
<td>-0.87329868</td>
<td>1.4472987</td>
</tr>
<tr>
<td>Study 5</td>
<td>0.052000</td>
<td>0.8640000</td>
<td>-1.64140888</td>
<td>1.7454089</td>
</tr>
<tr>
<td>Study 6</td>
<td>0.146000</td>
<td>0.2410000</td>
<td>-0.32635132</td>
<td>0.6183513</td>
</tr>
<tr>
<td>Study 7</td>
<td>0.562000</td>
<td>0.2520000</td>
<td>0.06808908</td>
<td>1.0559109</td>
</tr>
<tr>
<td>Study 8</td>
<td>0.904000</td>
<td>0.3180000</td>
<td>0.28073145</td>
<td>1.5272685</td>
</tr>
</tbody>
</table>
meta.ave.gen.cc  Confidence interval for an average effect size using a constant coefficient model

Description

Computes the estimate, standard error, and confidence interval for a weighted average effect from two or more studies using the constant coefficient (fixed-effect) meta-analysis model. The weighted average estimate will be biased regardless of number of studies or sample size per study and the actual confidence interval coverage probability can be much smaller than the specified confidence level when the true effect sizes are not identical across studies.

Usage

meta.ave.gen.cc(alpha, est, se, bystudy = TRUE)

Arguments

- alpha  alpha level for 1-alpha confidence
- est     vector of parameter estimates
- se      vector of standard errors
- bystudy logical to also return each study estimate (TRUE) or not

Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References

Examples

est <- c(.022, .751, .421, .287, .052, .146, .562, .904)
se <- c(.124, .464, .102, .592, .864, .241, .252, .318)
meta.ave.gen.cc(.05, est, se, bystudy = TRUE)

# Should return:
# Estimate    SE      LL       UL
# Average  0.3127916 0.06854394 0.17844794 0.4471352
# Study 1   0.0220000 0.12400000 -0.22103553 0.2650355
# Study 2   0.7510000 0.46400000 -0.15842329 1.6604233
# Study 3   0.4210000 0.10200000 0.22108367 0.6209163
# Study 4   0.2870000 0.59200000 -0.87329868 1.4472987
# Study 5   0.0520000 0.86400000 -1.64140888 1.7454089
# Study 6   0.1460000 0.24100000 -0.32635132 0.6183513
# Study 7   0.5620000 0.25200000 0.06808908 1.0559109
# Study 8   0.9040000 0.31800000 0.28073145 1.5272685

meta.ave.gen.rc

Confidence interval for an average effect size using a random coefficient model

Description

Computes the estimate, standard error, and confidence interval for a weighted average effect from multiple studies using the random coefficient (random-effects) meta-analysis model. An estimate of effect-size heterogeneity (tau-squared) is also computed. The random coefficient model assumes that the studies in the meta-analysis are a random sample from some definable superpopulation of studies. This assumption is very difficult to justify. The weighted average estimate will be biased regardless of number of studies or sample size per study and the actual confidence interval coverage probability can much smaller than the specified confidence level if the effect sizes are correlated with the weights. This method also assume that the true effects sizes in the superpopulation of studies have a normal distribution. A large number of studies, each with a large sample size, is required to assess the superpopulation normality assumption and to accurately estimate tau-squared. The traditional confidence interval for the population tau-squared is hypersensitive to very minor and difficult to detect violations of the superpopulation normality assumption.

Usage

meta.ave.gen.rc(alpha, est, se, bystudy = TRUE)

Arguments

alpha alpha level for 1-alpha confidence
est vector of parameter estimates
se vector of standard errors
bystudy logical to also return each study estimate (TRUE) or not
meta.ave.mean.ps

Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

```r
est <- c(.022, .751, .421, .287, .052, .146, .562, .904)
se <- c(.124, .464, .102, .592, .864, .241, .252, .318)
meta.ave.gen.rc(.05, est, se, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tau-squared</td>
<td>0.03772628</td>
<td>0.0518109</td>
<td>0.00000000</td>
<td>0.1392738</td>
</tr>
<tr>
<td>Average</td>
<td>0.35394806</td>
<td>0.1155239</td>
<td>0.1275258</td>
<td>0.5803788</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.02200000</td>
<td>0.1240000</td>
<td>-0.22103553</td>
<td>0.2650355</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.75100000</td>
<td>0.4640000</td>
<td>-0.15842329</td>
<td>1.6604233</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.42100000</td>
<td>0.1020000</td>
<td>0.22103553</td>
<td>0.6209163</td>
</tr>
<tr>
<td>Study 4</td>
<td>0.28700000</td>
<td>0.4640000</td>
<td>-0.87329868</td>
<td>1.4472987</td>
</tr>
<tr>
<td>Study 5</td>
<td>0.05200000</td>
<td>0.8640000</td>
<td>-1.64140888</td>
<td>1.7454089</td>
</tr>
<tr>
<td>Study 6</td>
<td>0.14600000</td>
<td>0.2410000</td>
<td>-0.32635132</td>
<td>0.6183513</td>
</tr>
<tr>
<td>Study 7</td>
<td>0.56200000</td>
<td>0.2520000</td>
<td>0.06889088</td>
<td>1.0559109</td>
</tr>
<tr>
<td>Study 8</td>
<td>0.90400000</td>
<td>0.3180000</td>
<td>0.28073145</td>
<td>1.5272685</td>
</tr>
</tbody>
</table>

---

**Confidence interval for an average mean difference from paired-samples studies**

Description

Computes the estimate, standard error, and confidence interval for an average mean difference from two or more paired-samples studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval for the average effect size. Equal variances within or across studies is not assumed.
Usage

```r
meta.ave.mean.ps(alpha, m1, m2, sd1, sd2, cor, n, bystudy = TRUE)
```

Arguments

- **alpha**: alpha level for 1-alpha confidence
- **m1**: vector of estimated means for measurement 1
- **m2**: vector of estimated means for measurement 2
- **sd1**: vector of estimated SDs for measurement 1
- **sd2**: vector of estimated SDs for measurement 2
- **cor**: vector of estimated correlations for paired measurements
- **n**: vector of sample sizes
- **bystudy**: logical to also return each study estimate (TRUE) or not

Value

A matrix. First row is the overall average estimate. If bystudy is TRUE also returns 1 row per study. The matrix has the following columns:

- **Estimate**: the estimated effect size
- **SE**: standard error
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval
- **df**: degrees of freedom

References


Examples

```r
m1 <- c(53, 60, 53, 57)
```

```r
m2 <- c(55, 62, 58, 61)
```

```r
sd1 <- c(4.1, 4.2, 4.5, 4.0)
```

```r
sd2 <- c(4.2, 4.7, 4.9, 4.8)
```

```r
cor <- c(.7, .7, .8, .85)
```

```r
n <- c(30, 50, 30, 70)
```

```r
meta.ave.mean.ps(.05, m1, m2, sd1, sd2, cor, n, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>-3.25</td>
<td>-3.739691</td>
<td>-2.7603091</td>
<td>112.347</td>
</tr>
<tr>
<td>Study 1</td>
<td>-2.00</td>
<td>-3.200836</td>
<td>-0.7991639</td>
<td>29.000</td>
</tr>
<tr>
<td>Study 2</td>
<td>-2.00</td>
<td>-2.988335</td>
<td>-1.0116648</td>
<td>49.000</td>
</tr>
<tr>
<td>Study 3</td>
<td>-5.00</td>
<td>-6.118973</td>
<td>-3.8810270</td>
<td>29.000</td>
</tr>
</tbody>
</table>
meta.ave.mean2

Confidence interval for an average mean difference from 2-group studies

Description

Computes the estimate, standard error, and confidence interval for an average mean difference from two or more 2-group studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence intervals. Equal variances within or across studies is not assumed.

Usage

meta.ave.mean2(alpha, m1, m2, sd1, sd2, n1, n2, bystudy = TRUE)

Arguments

alpha alpha level for 1-alpha confidence
m1 vector of estimated means for group 1
m2 vector of estimated means for group 2
sd1 vector of estimated SDs for group 1
sd2 vector of estimated SDs for group 2
n1 vector of group 1 sample sizes
n2 vector of group 2 sample sizes
bystudy logical to also return each study estimate (TRUE) or not

Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- df - degrees of freedom

References

meta.ave.meanratio.ps

Examples

\[ m1 \leftarrow c(7.4, 6.9) \]
\[ m2 \leftarrow c(6.3, 5.7) \]
\[ sd1 \leftarrow c(1.72, 1.53) \]
\[ sd2 \leftarrow c(2.35, 2.04) \]
\[ n1 \leftarrow c(40, 60) \]
\[ n2 \leftarrow c(40, 60) \]
\[ meta.ave.mean2(.05, m1, m2, sd1, sd2, n1, n2, bystudy = TRUE) \]

# Should return:
# Estimate  SE  LL  UL  df
# Average  1.15  0.2830183  0.5904369  1.709563  139.41053
# Study 1   1.10  0.4604590  0.1819748  2.018025  71.46729
# Study 2   1.20  0.3292036  0.5475574  1.852443  109.42136

meta.ave.meanratio.ps  Confidence interval for an average mean ratio from paired-samples studies

Description

Computes the estimate, standard error, and confidence interval for an geometric average mean ratio from two or more paired-samples studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval for the average effect size. Equal variances within or across studies is not assumed.

Usage

meta.ave.meanratio.ps(alpha, m1, m2, sd1, sd2, cor, n, bystudy = TRUE)

Arguments

alpha  alpha level for 1-alpha confidence
m1  vector of estimated means for measurement 1
m2  vector of estimated means for measurement 2
sd1  vector of estimated SDs for measurement 1
sd2  vector of estimated SDs for measurement 2
cor  vector of estimated correlations for paired measurements
n  vector of sample sizes
bystudy  logical to also return each study estimate (TRUE) or not
Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - the exponentiated estimate
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval
- df - degrees of freedom

Examples

```r
m1 <- c(53, 60, 53, 57)
m2 <- c(55, 62, 58, 61)
sd1 <- c(4.1, 4.2, 4.5, 4.0)
sd2 <- c(4.2, 4.7, 4.9, 4.8)
cor <- c(.7, .7, .8, .85)
n <- c(30, 50, 30, 70)
meta.ave.meanratio.ps(.05, m1, m2, sd1, sd2, cor, n, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th>#</th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>-0.05695120</td>
<td>0.004350863</td>
<td>-0.06558008</td>
<td>-0.04832231</td>
</tr>
<tr>
<td>Study 1</td>
<td>-0.03704127</td>
<td>0.010871086</td>
<td>-0.05927514</td>
<td>-0.01480740</td>
</tr>
<tr>
<td>Study 2</td>
<td>-0.03278982</td>
<td>0.008021952</td>
<td>-0.04891054</td>
<td>-0.01666911</td>
</tr>
<tr>
<td>Study 3</td>
<td>-0.09015110</td>
<td>0.009779919</td>
<td>-0.11015328</td>
<td>-0.07014892</td>
</tr>
<tr>
<td>Study 4</td>
<td>-0.06782260</td>
<td>0.004970015</td>
<td>-0.07773750</td>
<td>-0.05790769</td>
</tr>
<tr>
<td>exp(Estimate)</td>
<td>0.9446402</td>
<td>0.93565240</td>
<td>0.9528266</td>
<td>103.0256</td>
</tr>
<tr>
<td>exp(LL)</td>
<td>exp(UL)</td>
<td>exp(LL)</td>
<td>exp(UL)</td>
<td>df</td>
</tr>
<tr>
<td>Average</td>
<td>0.9446402</td>
<td>0.93565240</td>
<td>0.9528266</td>
<td>103.0256</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.9636364</td>
<td>0.9424474</td>
<td>0.9853017</td>
<td>29.0000</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.967419</td>
<td>0.9522663</td>
<td>0.9834691</td>
<td>49.0000</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.9137931</td>
<td>0.8956968</td>
<td>0.9322550</td>
<td>29.0000</td>
</tr>
<tr>
<td>Study 4</td>
<td>0.9344262</td>
<td>0.9252073</td>
<td>0.9437371</td>
<td>69.0000</td>
</tr>
</tbody>
</table>
```

Description

Computes the estimate, standard error, and confidence interval for an geometric average mean ratio from two or more 2-group studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence intervals. Equal variances within or across studies is not assumed.
Usage

`meta.ave.meanratio2(alpha, m1, m2, sd1, sd2, n1, n2, bystudy = TRUE)`

Arguments

- `alpha` alpha level for 1-alpha confidence
- `m1` vector of estimated means for group 1
- `m2` vector of estimated means for group 2
- `sd1` vector of estimated SDs for group 1
- `sd2` vector of estimated SDs for group 2
- `n1` vector of group 1 sample sizes
- `n2` vector of group 2 sample sizes
- `bystudy` logical to also return each study estimate (TRUE) or not

Value

A matrix. First row is the overall average estimate. If `bystudy` is TRUE also returns 1 row per study. The matrix has the following columns:

- `Estimate` - the estimated effect size
- `SE` - standard error
- `LL` - lower limit of the confidence interval
- `UL` - upper limit of the confidence interval
- `exp(Estimate)` - the exponentiated estimate
- `exp(LL)` - lower limit of the exponentiated confidence interval
- `exp(UL)` - upper limit of the exponentiated confidence interval
- `df` - degrees of freedom

References


Examples

```r
m1 <- c(7.4, 6.9)
m2 <- c(6.3, 5.7)
sd1 <- c(1.7, 1.5)
sd2 <- c(2.3, 2.0)
n1 <- c(40, 20)
n2 <- c(40, 20)
meta.ave.meanratio2(.05, m1, m2, sd1, sd2, n1, n2, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
<th>exp(Estimate)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```r
# Average 0.1759928 0.05738065 0.061437186 0.2905484 1.192429
# Study 1 0.1609304 0.06820167 0.024749712 0.2971110 1.174603
# Study 2 0.1910552 0.09229675 0.002986265 0.3791242 1.210526
# exp(LL) exp(UL) df
# Average 1.063364 1.337161 66.26499
# Study 1 1.025059 1.345965 65.69929
# Study 2 1.002991 1.461004 31.71341
```

---

### meta.ave.odds

Confidence interval for average odds ratio from 2-group studies

#### Description

Computes the estimate, standard error, and confidence interval for a geometric average odds ratio from two or more studies.

#### Usage

```r
text.meta.ave.odds(alpha, f1, f2, n1, n2, bystudy = TRUE)
```

#### Arguments

- `alpha`: alpha level for 1-alpha confidence
- `f1`: vector of group 1 event counts
- `f2`: vector of group 2 event counts
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `bystudy`: logical to also return each study estimate (TRUE) or not

#### Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- **Estimate**: the estimated effect size
- **SE**: standard error
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval
- **exp(Estimate)**: the exponentiated estimate
- **exp(LL)**: lower limit of the exponentiated confidence interval
- **exp(UL)**: upper limit of the exponentiated confidence interval
meta.ave.path

References


Examples

n1 <- c(204, 201, 932, 130, 77)
n2 <- c(106, 103, 415, 132, 83)
f1 <- c(24, 40, 93, 14, 5)
f2 <- c(12, 9, 28, 3, 1)
meta.ave.odds(.05, f1, f2, n1, n2, bystudy = TRUE)

# Should return:
# Estimate SE LL UL
# Average 0.86211102 0.2512852 0.36960107 1.3546210
# Study 1 0.02581353 0.3700520 -0.69947512 0.7511022
# Study 2 0.91410487 0.3830515 0.16333766 1.6648721
# Study 3 0.41496672 0.2226089 -0.02133877 0.8512722
# Study 4 1.52717529 0.6090858 0.33338907 2.7209615
# Study 5 1.42849472 0.9338907 -0.40425414 3.2612436
# exp(Estimate) exp(LL) exp(UL)
# Average 2.368155 1.4471572 3.875292
# Study 1 1.026150 0.4968460 2.119335
# Study 2 2.494541 1.1774342 5.284997
# Study 3 1.514320 0.9788873 2.342625
# Study 4 4.605150 1.3956902 15.194925
# Study 5 4.172414 0.6674745 26.081952

meta.ave.path

Confidence interval for an average slope coefficient in a general linear model or a path model.

Description

Computes the estimate, standard error, and confidence interval for an average slope coefficient in a general linear model (ANOVA, ANCOVA, multiple regression) or a path model from two or more studies.

Usage

meta.ave.path(alpha, n, slope, se, s, bystudy = TRUE)
Arguments

alpha  alpha level for 1-alpha confidence
n      vector of sample sizes
slope  vector of slope estimates
se     vector of slope standard errors
s      number of predictors of the response variable
bystudy logical to also return each study estimate (TRUE) or not

Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

Examples

```r
n <- c(75, 85, 250, 160)
slope <- c(1.57, 1.38, 1.08, 1.25)
se <- c(.658, .724, .307, .493)
meta.ave.path(.05, n, slope, se, 2, bystudy = TRUE)
```

# Should return:
# Estimate SE LL UL df
# Average 1.32 0.2844334 0.75994528 1.880055 263.1837
# Study 1 1.57 0.6580000 0.25830097 2.881699 72.0000
# Study 2 1.38 0.7240000 -0.06026664 2.820267 82.0000
# Study 3 1.08 0.3070000 0.47532827 1.684672 247.0000
# Study 4 1.25 0.4930000 0.27623174 2.223768 157.0000
```

Description

Computes the estimate, standard error, and confidence interval for an average point-biserial correlation from two or more studies. Two types of point-biserial correlations can be meta-analyzed. One type uses an unweighted variance and is appropriate in 2-group experimental designs. The other type uses a weighted variance and is appropriate in 2-group nonexperimental designs with simple random sampling within each group. This function requires all point-biserial correlations to be of the same type. Use the meta.ave.gen function to meta-analyze any combination of biserial correlation types.
Usage

meta.ave.pbcor(alpha, m1, m2, sd1, sd2, n1, n2, type, bystudy = TRUE)

Arguments

alpha  
alpha level for 1-alpha confidence
m1  
vector of estimated means for group 1
m2  
vector of estimated means for group 2
sd1  
vector of estimated SDs for group 1
sd2  
vector of estimated SDs for group 2
n1  
vector of group 1 sample sizes
n2  
vector of group 2 sample sizes
type  
• set to 1 for weighted variance
• set to 2 for unweighted variance
bystudy  
logical to also return each study estimate (TRUE) or not

Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

• Estimate - the estimated effect size
• SE - standard error
• LL - lower limit of the confidence interval
• UL - upper limit of the confidence interval

References


Examples

m1 <- c(21.9, 23.1, 19.8)
m2 <- c(16.1, 17.4, 15.0)
sd1 <- c(3.82, 3.95, 3.67)
sd2 <- c(3.21, 3.30, 3.02)
n1 <- c(40, 30, 24)
n2 <- c(40, 28, 25)
meta.ave.pbcor(.05, m1, m2, sd1, sd2, n1, n2, 2, bystudy = TRUE)

# Should return:
# Estimate SE LL UL
# Average 0.6159094 0.04363432 0.5230976 0.6942842
# Study 1 0.6349786 0.06316796 0.4842020 0.7370220
meta.ave.plot

Forest plot for average effect sizes

Description
Generates a forest plot to visualize effect sizes estimates and overall averages from the meta.ave functions in vcmeta. If the column exp(Estimate) is present, this function plots the exponentiated effect size and CI found in columns exp(Estimate), exp(LL), and exp(UL). Otherwise, this function plots the effect size and CI found in the columns Estimate, LL, and UL.

Usage
meta.ave.plot(
result,
reference_line = NULL,
diamond_height = 0.2,
ggtheme = ggplot2::theme_classic()
)

Arguments
result • a result matrix from any of the replicate functions in vcmeta
reference_line Optional x-value for a reference line. Only applies if focus is 'Difference' or 'Both'. Defaults to NULL, in which case a reference line is not drawn.
diamond_height • Optional height of the diamond representing average effect size. Only applies if focus is 'Average' or 'Both'. Defaults to 0.2
ggtheme • optional ggplot2 theme object; defaults to theme_classic()

Value
Returns a ggplot object. If stored, can be further customized via the ggplot API

Examples
# Plot results from meta.ave.mean2
m1 <- c(7.4, 6.9)
m2 <- c(6.3, 5.7)
sd1 <- c(1.72, 1.53)
sd2 <- c(2.35, 2.04)
n1 <- c(40, 60)
n2 <- c(40, 60)
result <- meta.ave.mean2(.05, m1, m2, sd1, sd2, n1, n2, bystudy = TRUE)
meta.ave.plot(result, reference_line = 0)
# Plot results from meta.ave.meanratio2
# Note that this plots the exponentiated effect size and CI
m1 <- c(53, 60, 53, 57)
m2 <- c(55, 62, 58, 61)
sd1 <- c(4.1, 4.2, 4.5, 4.0)
sd2 <- c(4.2, 4.7, 4.9, 4.8)
cor <- c(.7, .7, .8, .85)
n <- c(30, 50, 30, 70)
result <- meta.ave.meanratio.ps(.05, m1, m2, sd1, sd2, cor, n, bystudy = TRUE)
myplot <- meta.ave.plot(result, reference_line = 1)
myplot

# Change x-scale to log2
library(ggplot2)
myplot <- myplot + scale_x_continuous(
  trans = 'log2',
  limits = c(0.75, 1.25),
  name = "Estimated Ratio of Means, Log2 Scale"
)
myplot

---

meta.ave.prop.ps  
Confidence interval for an average proportion difference in paired-samples studies

Description

Computes the estimate, standard error, and confidence interval for an average proportion difference from two or more studies.

Usage

meta.ave.prop.ps(alpha, f11, f12, f21, f22, bystudy = TRUE)

Arguments

alpha  
alpha level for 1-alpha confidence

f11  
vector of frequencies in cell 1,1

f12  
vector of frequencies in cell 1,2

f21  
vector of frequencies in cell 2,1

f22  
vector of frequencies in cell 2,2

bystudy  
logical to also return each study estimate (TRUE) or not
meta.ave.prop2

**Value**

Returns a matrix. The first row is the average estimate across all studies. If `bystudy` is true, there is 1 additional row for each study. The matrix has the following columns:

- **Estimate**: the estimated effect size
- **SE**: standard error
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval

**References**


**Examples**

```r
f11 <- c(17, 28, 19)
f12 <- c(43, 56, 49)
f21 <- c(3, 5, 5)
f22 <- c(37, 54, 39)
meta.ave.prop.ps(.05, f11, f12, f21, f22, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>0.380573</td>
<td>0.0300016</td>
<td>0.3221581</td>
<td>0.4397565</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.3921569</td>
<td>0.05573055</td>
<td>0.2829270</td>
<td>0.5013867</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.3517241</td>
<td>0.04629537</td>
<td>0.2609869</td>
<td>0.4424614</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.3859649</td>
<td>0.05479300</td>
<td>0.2785726</td>
<td>0.4933572</td>
</tr>
</tbody>
</table>
```

---

**Description**

Computes the estimate, standard error, and confidence interval for an average proportion difference from two or more studies.

**Usage**

```r
meta.ave.prop2(alpha, f1, f2, n1, n2, bystudy = TRUE)
```
**meta.ave.prop2**

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `f1`: vector of group 1 event counts
- `f2`: vector of group 2 event counts
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `bystudy`: logical to also return each study estimate (TRUE) or not

**Value**

Returns a matrix. The first row is the average estimate across all studies. If `bystudy` is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

**References**


**Examples**

```r
n1 <- c(204, 201, 932, 130, 77)
n2 <- c(106, 103, 415, 132, 83)
f1 <- c(24, 40, 93, 14, 5)
f2 <- c(12, 9, 28, 3, 1)
meta.ave.prop2(.05, f1, f2, n1, n2, bystudy = TRUE)
```

```
# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>0.0567907589</td>
<td>0.01441216</td>
<td>2.854345e-02</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.0009888529</td>
<td>0.03870413</td>
<td>-7.486985e-02</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.1067323481</td>
<td>0.04018243</td>
<td>2.797623e-02</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.0310980338</td>
<td>0.01587717</td>
<td>-2.064379e-02</td>
</tr>
<tr>
<td>Study 4</td>
<td>0.0837856174</td>
<td>0.03129171</td>
<td>2.245499e-02</td>
</tr>
<tr>
<td>Study 5</td>
<td>0.0524199553</td>
<td>0.03403926</td>
<td>-1.429577e-02</td>
</tr>
</tbody>
</table>
```

Confidence interval for an average proportion ratio from 2-group studies

Description
Computes the estimate, standard error, and confidence interval for a geometric average proportion ratio from two or more studies.

Usage
meta.ave.propratio2(alpha, f1, f2, n1, n2, bystudy = TRUE)

Arguments
- alpha: alpha level for 1-alpha confidence
- f1: vector of group 1 event counts
- f2: vector of group 2 event counts
- n1: vector of group 1 sample sizes
- n2: vector of group 2 sample sizes
- bystudy: logical to also return each study estimate (TRUE) or not

Value
Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - the exponentiated estimate
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval

References
**Examples**

```r
n1 <- c(204, 201, 932, 130, 77)
n2 <- c(106, 103, 415, 132, 83)
f1 <- c(24, 40, 93, 14, 5)
f2 <- c(12, 9, 28, 3, 1)
meta.ave.propratio2(.05, f1, f2, n1, n2, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>0.847056</td>
<td>0.252874</td>
<td>0.351431</td>
<td>1.342680</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.036042</td>
<td>0.329740</td>
<td>0.610236</td>
<td>0.682322</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.810089</td>
<td>0.344200</td>
<td>0.135468</td>
<td>1.484710</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.387483</td>
<td>0.206522</td>
<td>-0.017308</td>
<td>0.792245</td>
</tr>
<tr>
<td>Study 4</td>
<td>1.493168</td>
<td>0.602329</td>
<td>0.312637</td>
<td>2.573712</td>
</tr>
<tr>
<td>Study 5</td>
<td>1.508512</td>
<td>0.982842</td>
<td>-0.417823</td>
<td>3.434846</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>exp(Estimate)</th>
<th>exp(LL)</th>
<th>exp(UL)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>2.332769</td>
<td>1.421100</td>
<td>3.829294</td>
</tr>
<tr>
<td>Study 1</td>
<td>1.036700</td>
<td>0.543222</td>
<td>1.978466</td>
</tr>
<tr>
<td>Study 2</td>
<td>2.248109</td>
<td>1.145073</td>
<td>4.413686</td>
</tr>
<tr>
<td>Study 3</td>
<td>1.473246</td>
<td>0.982840</td>
<td>2.208350</td>
</tr>
<tr>
<td>Study 4</td>
<td>4.451175</td>
<td>1.367087</td>
<td>14.493677</td>
</tr>
<tr>
<td>Study 5</td>
<td>4.520000</td>
<td>0.658478</td>
<td>31.026662</td>
</tr>
</tbody>
</table>
```

---

**meta.ave.semipart**

*Confidence interval for an average semipartial correlation*

**Description**

Computes the estimate, standard error, and confidence interval for an average semipartial correlation from two or more studies.

**Usage**

```r
meta.ave.semipart(alpha, n, cor, r2, bystudy = TRUE)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `n`: vector of sample sizes
- `cor`: vector of estimated semipartial correlations
- `r2`: vector of squared multiple correlations for full model
- `bystudy`: logical to also return each study estimate (TRUE) or not
Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

Examples

```r
n <- c(128, 97, 210, 217)
cor <- rbind(.35, .41, .44, .39)
r2 <- rbind(.29, .33, .36, .39)
meta.ave.semipart(.05, n, cor, r2, bystudy = TRUE)
```

```
# Should return:
# Estimate SE LL UL
# Average 0.3975 0.03221240 0.3325507 0.4586965
# Study 1 0.3500 0.07175200 0.2023485 0.4820930
# Study 2 0.4100 0.07886080 0.2447442 0.5521076
# Study 3 0.4400 0.05146694 0.3338366 0.5351410
# Study 4 0.3900 0.05085271 0.2860431 0.4848830
```

Description

Computes the estimate, standard error, and confidence interval for an average slope coefficient in a simple linear regression model from two or more studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval.

Usage

```r
meta.ave.slope(alpha, n, cor, sdy, sdx, bystudy = TRUE)
```

Arguments

- `alpha` alpha level for 1-alpha confidence
- `n` vector of sample sizes
- `cor` vector of estimated correlations
- `sdy` vector of estimated SDs of y
- `sdx` vector of estimated SDs of x
- `bystudy` logical to also return each study estimate (TRUE) or not
**Value**

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- df - degrees of freedom

**Examples**

```r
n <- c(45, 85, 50, 60)
cor <- c(.24, .35, .16, .20)
sdy <- c(12.2, 14.1, 11.7, 15.9)
sdx <- c(1.34, 1.87, 2.02, 2.37)
meta.ave.slope(.05, n, cor, sdy, sdx, bystudy = TRUE)
```

# Should return:

```
#    Estimate    SE    LL    UL    df
# Average  1.7731542 0.4755417 0.8335021 2.712806 149.4777
# Study 1  2.1850746 1.3084468 -0.4536599 4.823809 43.0000
# Study 2  2.6390374 0.7262491 1.1945573 4.083518 83.0000
# Study 3  0.9267327 0.8146126 -0.7111558 2.564621 48.0000
# Study 4  1.3417722 0.8456799 -0.3510401 3.034584 58.0000
```

**meta.ave.spear**

*Confidence interval for an average Spearman correlation*

**Description**

Computes the estimate, standard error, and confidence interval for an average Spearman correlation from two or more studies. The Spearman correlation is preferred to the Pearson correlation if the relation between the two quantitative variables is monotonic rather than linear or if the bivariate normality assumption is not plausible.

**Usage**

```r
meta.ave.spear(alpha, n, cor, bystudy = TRUE)
```

**Arguments**

- **alpha** - alpha level for 1-alpha confidence
- **n** - vector of sample sizes
- **cor** - vector of estimated Spearman correlations
- **bystudy** - logical to also return each study estimate (TRUE) or not
Value

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

```r
n <- c(150, 200, 300, 200, 350)
cor <- c(.14, .29, .16, .21, .23)
meta.ave.spear(.05, n, cor, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>0.206</td>
<td>0.02944265</td>
<td>0.14763960</td>
<td>0.2629309</td>
</tr>
<tr>
<td>Study 1</td>
<td>0.140</td>
<td>0.08031750</td>
<td>-0.02151639</td>
<td>0.2943944</td>
</tr>
<tr>
<td>Study 2</td>
<td>0.290</td>
<td>0.06492643</td>
<td>0.15476515</td>
<td>0.4145671</td>
</tr>
<tr>
<td>Study 3</td>
<td>0.160</td>
<td>0.05635101</td>
<td>0.04689807</td>
<td>0.2690514</td>
</tr>
<tr>
<td>Study 4</td>
<td>0.210</td>
<td>0.06776195</td>
<td>0.07187439</td>
<td>0.3402225</td>
</tr>
<tr>
<td>Study 5</td>
<td>0.230</td>
<td>0.05069710</td>
<td>0.12690280</td>
<td>0.3281809</td>
</tr>
</tbody>
</table>

---

**meta.ave.stdmean.ps**

Confidence interval for an average standardized mean difference from paired-samples studies

Description

Computes the estimate, standard error, and confidence interval for an average standardized mean difference from two or more paired-samples studies. Unweighted variances and single group variance are options for the standardizer. Equal variances within or across studies is not assumed.

Usage

```r
meta.ave.stdmean.ps(alpha, m1, m2, sd1, sd2, cor, n, stdzr, bystudy = TRUE)
```
Arguments

- **alpha**: alpha level for 1-alpha confidence
- **m1**: vector of estimated means for measurement 1
- **m2**: vector of estimated means for measurement 2
- **sd1**: vector of estimated SDs for measurement 1
- **sd2**: vector of estimated SDs for measurement 2
- **cor**: vector of estimated correlations for paired measurements
- **n**: vector of sample sizes
- **stdzr**: set to 0 for square root unweighted average variance standardizer, set to 1 for group 1 SD standardizer, set to 2 for group 2 SD standardizer
- **bystudy**: logical to also return each study estimate (TRUE) or not

Value

A matrix. First row is the overall average estimate. If bystudy is TRUE also returns 1 row per study. The matrix has the following columns:

- **Estimate**: the estimated effect size
- **SE**: standard error
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval

References


Examples

```r
m1 <- c(23.9, 24.1)
m2 <- c(25.1, 26.9)
sd1 <- c(1.76, 1.58)
sd2 <- c(2.01, 1.76)
cor <- c(.78, .84)
n <- c(25, 30)
meta.ave.stdmean.ps(.05, m1, m2, sd1, sd2, cor, n, 1, bystudy = TRUE)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>-1.1931045</td>
<td>0.1568034</td>
<td>-1.500433</td>
</tr>
<tr>
<td>Study 1</td>
<td>-0.6818182</td>
<td>0.1773785</td>
<td>-1.029474</td>
</tr>
<tr>
<td>Study 2</td>
<td>-1.7721519</td>
<td>0.2586234</td>
<td>-2.279044</td>
</tr>
</tbody>
</table>
Description

Computes the estimate, standard error, and confidence interval for an average standardized mean difference from two or more 2-group studies. Unweighted variances, weighted variances, and single group variance are options for the standardizer. Equal variances within or across studies is not assumed.

Usage

`meta.ave.stdmean2(alpha, m1, m2, sd1, sd2, n1, n2, stdzr, bystudy = TRUE)`

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `m1`: vector of estimated means for group 1
- `m2`: vector of estimated means for group 2
- `sd1`: vector of estimated SDs for group 1
- `sd2`: vector of estimated SDs for group 2
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `stdzr`: • set to 0 for square root unweighted average variance standardizer
  • set to 1 for group 1 SD standardizer
  • set to 2 for group 2 SD standardizer
  • set to 3 for square root weighted average variance standardizer
- `bystudy`: logical to also return each study estimate (TRUE) or not

Value

A matrix. First row is the overall average estimate. If `bystudy` is TRUE also returns 1 row per study. The matrix has the following columns:

- Estimate - the estimated effect size
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References

Examples

```r
m1 <- c(21.9, 23.1, 19.8)
m2 <- c(16.1, 17.4, 15.0)
sd1 <- c(3.82, 3.95, 3.67)
sd2 <- c(3.21, 3.30, 3.02)
n1 <- c(40, 30, 24)
n2 <- c(40, 28, 25)
meta.ave.stdmean2(.05, m1, m2, sd1, sd2, n1, n2, 0, bystudy = TRUE)
```

# Should return:
# Estimate  SE    LL     UL
# Average  1.526146 0.1734341 1.1862217 1.866071
# Study 1  1.643894 0.2629049 1.1286100 2.159178
# Study 2  1.566132 0.3056278 0.9671126 2.165152
# Study 3  1.428252 0.3289179 0.7835848 2.072919
```

---

**meta.ave.var**

Confidence interval for an average variance

**Description**

Computes the estimate and confidence interval for an average variance from two or more studies.

**Usage**

`meta.ave.var(alpha, var, n, bystudy = TRUE)`

**Arguments**

- `alpha` alpha level for 1-alpha confidence
- `var` vector of sample variances
- `n` vector of sample sizes
- `bystudy` logical to also return each study estimate (TRUE) or not

**Value**

Returns a matrix. The first row is the average estimate across all studies. If bystudy is true, there is 1 additional row for each study. The matrix has the following columns:

- Estimate - the estimated variance
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
Examples

```r
var <- c(26.63, 22.45, 34.12)
n <- c(40, 30, 50)
meta.ave.var(.05, var, n, bystudy = TRUE)
```

# Should return:
# Estimate LL UL
# Average 27.73333 21.45679 35.84589
# Study 1 26.63000 17.86939 43.90614
# Study 2 22.45000 14.23923 40.57127
# Study 3 34.12000 23.80835 52.98319

---

**meta.chitest**

*Computes a chi-square test of effect-size homogeneity*

**Description**

Computes a chi-square test of effect size homogeneity and p-value using effect-size estimates and their standard errors from two or more studies. This test should not be used to justify the use of a constant coefficient (fixed-effect) meta-analysis. This test can be used to justify the estimation of an average effect size in a varying coefficient model.

**Usage**

```r
meta.chitest(est, se)
```

**Arguments**

- `est` vector of effect-size estimates
- `se` vector of effect-size standard errors

**Value**

Returns a one-row matrix:

- `Q` - chi-square test statistic
- `df` - degrees of freedom
- `p` - p-value

**References**

Examples

```r
est <- c(.297, .324, .281, .149)
se <- c(.082, .051, .047, .094)
meta.chitest(est, se)

# Should return:
#       Q   df         p
# 2.706526 3 0.4391195
```

---

**meta.lc.agree**

Confidence interval for a linear contrast of G-index coefficients

**Description**

Computes the estimate, standard error, and confidence interval for a linear contrast of G-index of agreement coefficients from two or more studies. This function assumes that two raters each provide a dichotomous rating for a sample of objects.

**Usage**

```r
meta.lc.agree(alpha, f11, f12, f21, f22, v)
```

**Arguments**

- **alpha**: alpha level for 1-alpha confidence
- **f11**: vector of frequencies in cell 1,1
- **f12**: vector of frequencies in cell 1,2
- **f21**: vector of frequencies in cell 2,1
- **f22**: vector of frequencies in cell 2,2
- **v**: vector of contrast coefficients

**Value**

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of linear contrast
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
Examples

```r
f11 <- c(17, 28, 19)
f12 <- c(43, 56, 49)
f21 <- c(3, 5, 5)
f22 <- c(37, 54, 39)
v <- c(.5, .5, -1)
meta.lc.agree(.05, f11, f12, f21, f22, v)
```

# Should return:
# Estimate SE LL UL
# Contrast 0.07692783 0.1138407 -0.1461958 0.3000515

---

**meta.lc.gen**

Confidence interval for a linear contrast of effect sizes

**Description**

Computes the estimate, standard error, and confidence interval for a linear contrast of any type of effect size from two or more studies.

**Usage**

```r
meta.lc.gen(alpha, est, se, v)
```

**Arguments**

- **alpha**: alpha level for 1-alpha confidence
- **est**: vector of parameter estimates
- **se**: vector of standard errors
- **v**: vector of contrast coefficients

**Value**

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of linear contrast
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
Examples

```r
est <- c(.55, .59, .44, .48, .26, .19)
se <- c(.054, .098, .029, .084, .104, .065)
v <- c(.5, .5, -.25, -.25, -.25, -.25)
meta.lc.gen(.05, est, se, v)
```

# Should return:
# Estimate SE LL UL
# Contrast 0.2275 0.06755461 0.0950954 0.3599046

---

**meta.lc.mean.ps**

*Confidence interval for a linear contrast of mean differences from paired-samples studies*

**Description**

Computes the estimate, standard error, and confidence interval for a linear contrast of paired-samples mean differences from two or more studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval. Equal variances within or across studies is not assumed.

**Usage**

```r
meta.lc.mean.ps(alpha, m1, m2, sd1, sd2, cor, n, v)
```

**Arguments**

- **alpha**: alpha level for 1-alpha confidence
- **m1**: vector of estimated means for group 1
- **m2**: vector of estimated means for group 2
- **sd1**: vector of estimated SDs for group 1
- **sd2**: vector of estimated SDs for group 2
- **cor**: vector of estimated correlations for paired measurements
- **n**: vector of sample sizes
- **v**: vector of contrast coefficients

**Value**

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- **Estimate** - estimate of linear contrast
- **SE** - standard error
- **LL** - lower limit of the confidence interval
- **UL** - upper limit of the confidence interval
- **df** - degrees of freedom
References


Examples

```r
m1 <- c(53, 60, 53, 57)
m2 <- c(55, 62, 58, 61)
sd1 <- c(4.1, 4.2, 4.5, 4.0)
sd2 <- c(4.2, 4.7, 4.9, 4.8)
cor <- c(.7, .7, .8, .85)
n <- c(30, 50, 30, 70)
v <- c(.5, .5, -.5, -.5)
meta.lc.mean.ps(.05, m1, m2, sd1, sd2, cor, n, v)
```

# Should return:

```
# Estimate    SE   LL    UL   df
# Contrast      2.5 0.4943114 1.520618 3.479382 112.347
```

---

**meta.lc.mean1**

*Confidence interval a for a linear contrast of means*

**Description**

Computes the estimate, standard error, and confidence interval for a linear contrast of means from two or more studies. This function will use either an unequal variance (recommended) or an equal variance method. A Satterthwaite adjustment to the degrees of freedom is used with the unequal variance method.

**Usage**

```r
meta.lc.mean1(alpha, m, sd, n, v, eqvar = FALSE)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `m`: vector of estimated means
- `sd`: vector of estimated standard deviations
- `n`: vector of sample sizes
- `v`: vector of contrast coefficients
- `eqvar`: 
  - FALSE for unequal variance method
  - TRUE for equal variance method
meta.lc.mean2

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of linear contrast
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- df - degrees of freedom

References


Examples

```r
m <- c(33.5, 37.9, 38.0, 44.1)
sd <- c(3.84, 3.84, 3.65, 4.98)
n <- c(10, 10, 10, 10)
v <- c(.5, .5, -.5, -.5)
meta.lc.mean1(.05, m, sd, n, v, eqvar = FALSE)
```

# Should return:
# Estimate    SE    LL    UL    df
# Contrast    -5.35 1.300136 -7.993583 -2.706417 33.52169

---

meta.lc.mean2

Confidence interval for a linear contrast of mean differences from 2-group studies

Description

Computes the estimate, standard error, and confidence interval for a linear contrast of 2-group mean differences from two or more studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval. Equal variances within or across studies is not assumed.

Usage

```r
meta.lc.mean2(alpha, m1, m2, sd1, sd2, n1, n2, v)
```
Arguments

- **alpha**: alpha level for 1-alpha confidence
- **m1**: vector of estimated means for group 1
- **m2**: vector of estimated means for group 2
- **sd1**: vector of estimated SDs for group 1
- **sd2**: vector of estimated SDs for group 2
- **n1**: vector of group 1 sample sizes
- **n2**: vector of group 2 sample sizes
- **v**: vector of contrast coefficients

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- **Estimate**: estimate of the linear contrast
- **SE**: standard error
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval
- **df**: degrees of freedom

References


Examples

```r
m1 <- c(45.1, 39.2, 36.3, 34.5)
m2 <- c(30.0, 35.1, 35.3, 36.2)
sd1 <- c(10.7, 10.5, 9.4, 11.5)
sd2 <- c(12.3, 12.0, 10.4, 9.6)
n1 <- c(40, 20, 50, 25)
n2 <- c(40, 20, 48, 26)
v <- c(.5, .5, -.5, -.5)
meta.lc.mean2(.05, m1, m2, sd1, sd2, n1, n2, v)

# Should return:
# Estimate SE  LL  UL  df
# Contrast  9.95 2.837787 4.343938 15.55606 153.8362
```
Description

Computes the estimate, standard error, and confidence interval for a log-linear contrast of paired-sample mean ratios from two or more studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval. Equal variances within or across studies is not assumed.

Usage

```
meta.lc.meanratio.ps(alpha, m1, m2, sd1, sd2, cor, n, v)
```

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `m1`: vector of estimated means for group 1
- `m2`: vector of estimated means for group 2
- `sd1`: vector of estimated SDs for group 1
- `sd2`: vector of estimated SDs for group 2
- `cor`: vector of estimated correlations for paired measurements
- `n`: vector of sample sizes
- `v`: vector of contrast coefficients

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of log-linear contrast
- SE - standard error of log-linear contrast
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - exponentiated estimate of log-linear function
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval
- df - degrees of freedom

References

Examples

```r
m1 <- c(53, 60, 53, 57)
m2 <- c(55, 62, 58, 61)
sd1 <- c(4.1, 4.2, 4.5, 4.0)
sd2 <- c(4.2, 4.7, 4.9, 4.8)
cor <- c(.7, .7, .8, .85)
n <- c(30, 50, 30, 70)
v <- c(.5, .5, -.5, -.5)
meta.lc.meanratio.ps(.05, m1, m2, sd1, sd2, cor, n, v)
```

# Should return:
# Contrast Estimate SE LL UL exp(Estimate)
# Contrast 0.040713 0.008701725 0.02681353 0.06132907 1.045057
# Contrast 1.027176 1.063249 103.0256

---

**Description**

Computes the estimate, standard error, and confidence interval for a log-linear contrast of 2-group mean ratio from two or more studies. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence interval. Equal variances within or across studies is not assumed.

**Usage**

```r
meta.lc.meanratio2(alpha, m1, m2, sd1, sd2, n1, n2, v)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `m1`: vector of estimated means for group 1
- `m2`: vector of estimated means for group 2
- `sd1`: vector of estimated SDs for group 1
- `sd2`: vector of estimated SDs for group 2
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `v`: vector of contrast coefficients
Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of log-linear contrast
- SE - standard error of log-linear contrast
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - the exponentiated estimate
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval
- df - degrees of freedom

References


Examples

```r
m1 <- c(45.1, 39.2, 36.3, 34.5)
m2 <- c(30.0, 35.1, 35.3, 36.2)
sd1 <- c(10.7, 10.5, 9.4, 11.5)
sd2 <- c(12.3, 12.0, 10.4, 9.6)
n1 <- c(40, 20, 50, 25)
n2 <- c(40, 20, 48, 26)
v <- c(.5, .5, -.5, -.5)
meta.lc.meanratio2(.05, m1, m2, sd1, sd2, n1, n2, v)
# Should return:
# Estimate  SE    LL    UL  exp(Estimate)
# Contrast 0.2691627 0.07959269 0.1119191 0.4264064 1.308868
# exp(LL)  exp(UL) df
# Contrast 1.118422 1.531743 152.8665
```

**meta.lc.odds**

Confidence interval for a log-linear contrast of odds ratios

Description

Computes the estimate, standard error, and confidence interval for an exponentiated log-linear contrast of odds ratios from two or more studies.
Usage

meta.lc.odds(alpha, f1, f2, n1, n2, v)

Arguments

alpha | alpha level for 1-alpha confidence
f1    | vector of group 1 event counts
f2    | vector of group 2 event counts
n1    | vector of group 1 sample sizes
n2    | vector of group 2 sample sizes
v     | vector of contrast coefficients

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of log-linear contrast
- SE - standard error of log-linear contrast
- exp(Estimate) - the exponentiated contrast
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval

References


Examples

```r
n1 <- c(50, 150, 150)
f1 <- c(16, 50, 25)
n2 <- c(50, 150, 150)
f2 <- c(7, 15, 20)
v <- c(1, -1, 0)
meta.lc.odds(.05, f1, f2, n1, n2, v)
```

# Should return:
# Estimate      SE    exp(Estimate)    exp(LL)    exp(UL)
# Contrast     -0.4596883 0.5895438     0.6314805  0.1988563  2.005305
Confidence interval for a linear contrast of proportion differences in paired-samples studies

Description

Computes the estimate, standard error, and confidence interval for a linear contrast of paired-samples group proportion differences from two or more studies.

Usage

\[
\text{meta.lc.prop.ps}(\alpha, f11, f12, f21, f22, v)
\]

Arguments

- \(\alpha\): alpha level for 1-\(\alpha\) confidence
- \(f11\): vector of frequencies in cell 1,1
- \(f12\): vector of frequencies in cell 1,2
- \(f21\): vector of frequencies in cell 2,1
- \(f22\): vector of frequencies in cell 2,2
- \(v\): vector of contrast coefficients

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of linear contrast
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

\[
f11 <- c(17, 28, 19)\]
\[
f12 <- c(43, 56, 49)\]
\[
f21 <- c(3, 5, 5)\]
\[
f22 <- c(37, 54, 39)\]
\[
v <- c(.5, .5, -1)\]
\[
\text{meta.lc.prop.ps}(.05, f11, f12, f21, f22, v)
\]
# Should return:
# Estimate   SE    LL    UL
# Contrast  -0.06511285 -0.1419817 0.113256

meta.lc.prop1

Confidence interval for a linear contrast of proportions.

Description

Computes the estimate, standard error, and an adjusted Wald confidence interval for a linear contrast of proportions from two or more studies.

Usage

meta.lc.prop1(alpha, f, n, v)

Arguments

alpha alpha level for 1-alpha confidence
f vector of frequency counts
n vector of sample sizes
v vector of contrast coefficients

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of linear contrast
- SE - standard error
- LL - lower limit of the adjusted Wald confidence interval
- UL - upper limit of the adjusted Wald confidence interval

References

Examples

```r
f <- c(26, 24, 38)
n <- c(60, 60, 60)
v <- c(-.5, -.5, 1)
meta.lc.prop1(.05, f, n, v)
```

# Should return:

```
#             Estimate    SE    LL    UL
# Contrast 0.2119565 0.07602892 0.06294259 0.3609705
```

---

**meta.lc.prop2**

*Confidence interval for a linear contrast of proportion differences in 2-group studies*

**Description**

Computes the estimate, standard error, and confidence interval for a linear contrast of 2-group proportion differences from two or more studies.

**Usage**

```r
meta.lc.prop2(alpha, f1, f2, n1, n2, v)
```

**Arguments**

- **alpha**: alpha level for 1-alpha confidence
- **f1**: vector of group 1 event counts
- **f2**: vector of group 2 event counts
- **n1**: vector of group 1 sample sizes
- **n2**: vector of group 2 sample sizes
- **v**: vector of contrast coefficients

**Value**

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of the linear contrast
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
References


Examples

```r	n1 <- c(50, 150, 150)
n2 <- c(50, 150, 150)
f1 <- c(16, 50, 25)
f2 <- c(7, 15, 20)
v <- c(1, -1, 0)
meta.lc.propratio2(.05, f1, f2, n1, n2, v)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.05466931</td>
<td>0.09401019</td>
<td>-0.2389259</td>
<td>0.1295873</td>
</tr>
</tbody>
</table>

---

**meta.lc.propratio2**

Confidence interval for a log-linear contrast of proportion ratios from 2-group studies

Description

Computes the estimate, standard error, and confidence interval for an exponentiated log-linear contrast of 2-group proportion ratios from two or more studies.

Usage

```r
meta.lc.propratio2(alpha, f1, f2, n1, n2, v)
```

Arguments

- `alpha` alpha level for 1-alpha confidence
- `f1` vector of group 1 event counts
- `f2` vector of group 2 event counts
- `n1` vector of group 1 sample sizes
- `n2` vector of group 2 sample sizes
- `v` vector of contrast coefficients
Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of log-linear contrast
- SE - standard error of log-linear contrast
- exp(Estimate) - the exponentiated contrast
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval

References


Examples

```r
n1 <- c(50, 150, 150)
f1 <- c(16, 50, 25)
n2 <- c(50, 150, 150)
f2 <- c(7, 15, 20)
v <- c(1, -1, 0)
meta.lc.propratio2(.05, f1, f2, n1, n2, v)
```

# Should return:
## Estimate SE exp(Estimate) exp(LL) exp(UL)
# Contrast -0.3853396 0.4828218 0.6802196 0.2640405 1.752378

---

**meta.lc.stdmean.ps**

Confidence interval for a linear contrast of standardized mean differences from paired-samples studies

### Description

Computes the estimate, standard error, and confidence interval for a linear contrast of paired-samples standardized mean differences from two or more studies. Equal variances within or across studies is not assumed.

### Usage

```r
meta.lc.stdmean.ps(alpha, m1, m2, sd1, sd2, cor, n, v, stdzr)
```
Arguments

- **alpha**: alpha level for 1-alpha confidence
- **m1**: vector of estimated means for group 1
- **m2**: vector of estimated means for group 2
- **sd1**: vector of estimated SDs for group 1
- **sd2**: vector of estimated SDs for group 2
- **cor**: vector of estimated correlations for paired measurements
- **n**: vector of sample sizes
- **v**: vector of contrast coefficients
- **stdzr**: set to 0 for square root unweighted average variance standardizer, set to 1 for group 1 SD standardizer, set to 2 for group 2 SD standardizer

Value

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- **Estimate**: estimate of linear contrast
- **SE**: standard error
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval

References


Examples

```r
m1 <- c(53, 60, 53, 57)
m2 <- c(55, 62, 58, 61)
sd1 <- c(4.1, 4.2, 4.5, 4.0)
sd2 <- c(4.2, 4.7, 4.9, 4.8)
cor <- c(.7, .7, .8, .85)
n <- c(30, 50, 30, 70)
v <- c(.5, .5, -.5, -.5)
meta.lc.stdmean.ps(.05, m1, m2, sd1, sd2, cor, n, v, 0)
```

# Should return:
# | Estimate | SE    | LL    | UL    |
# |----------|-------|-------|-------|
# | 0.5127577| 0.1392232| 0.2398851| 0.7856302 |
**meta.lc.stdmean2**

*Confidence interval for a linear contrast of standardized mean differences from 2-group studies*

**Description**

Computes the estimate, standard error, and confidence interval for a linear contrast of 2-group standardized mean differences from two or more studies. Equal variances within or across studies is not assumed. Use the square root average variance standardizer (stdzr = 0) for 2-group experimental designs. Use the square root weighted variance standardizer (stdzr = 3) for 2-group nonexperimental designs with simple random sampling. The stdzr = 1 and stdzr = 2 options can be used with either 2-group experimental or nonexperimental designs.

**Usage**

`meta.lc.stdmean2(alpha, m1, m2, sd1, sd2, n1, n2, v, stdzr)`

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `m1`: vector of estimated means for group 1
- `m2`: vector of estimated means for group 2
- `sd1`: vector of estimated SDs for group 1
- `sd2`: vector of estimated SDs for group 2
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `v`: vector of contrast coefficients
- `stdzr`: set to 0 for square root unweighted average variance standardizer
  - set to 1 for group 1 SD standardizer
  - set to 2 for group 2 SD standardizer
  - set to 3 for square root weighted average variance standardizer

**Value**

Returns 1-row matrix with the estimate for the contrast. The matrix has the following columns:

- Estimate - estimate of linear contrast
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
References


Examples

m1 <- c(45.1, 39.2, 36.3, 34.5)
m2 <- c(30.0, 35.1, 35.3, 36.2)
sd1 <- c(10.7, 10.5, 9.4, 11.5)
sd2 <- c(12.3, 12.0, 10.4, 9.6)
n1 <- c(40, 20, 50, 25)
n2 <- c(40, 20, 48, 26)
v <- c(0.5, 0.5, 0.5, 0.5)
meta.lc.stdmean2(.05, m1, m2, sd1, sd2, n1, n2, v, 0)

# Should return:
# Estimate SE LL UL
# Contrast 0.8557914 0.2709192 0.3247995 1.386783

---

meta.lm.agree

Meta-regression analysis for G agreement indices

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a G-index of agreement. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

Usage

meta.lm.agree(alpha, f11, f12, f21, f22, X)

Arguments

alpha alpha level for 1-alpha confidence
f11 vector of frequencies in cell 1,1
f12 vector of frequencies in cell 1,2
f21 vector of frequencies in cell 2,1
f22 vector of frequencies in cell 2,2
X matrix of predictor values
**Value**

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- **Estimate** - OLS estimate
- **SE** - standard error
- **z** - z-value
- **p** - p-value
- **LL** - lower limit of the confidence interval
- **UL** - upper limit of the confidence interval

**Examples**

```r
f11 <- c(40, 20, 25, 30)
f12 <- c(3, 2, 2, 1)
f21 <- c(7, 6, 8, 6)
f22 <- c(26, 25, 13, 25)
x1 <- c(1, 1, 4, 6)
x2 <- c(1, 1, 0, 0)
X <- matrix(cbind(x1, x2), 4, 2)
meta.lm.agree(.05, f11, f12, f21, f22, X)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>0.1904762</td>
<td>0.38772858</td>
<td>0.4912617</td>
<td>-0.56945786</td>
<td>0.9504102</td>
</tr>
<tr>
<td>b1</td>
<td>0.0952381</td>
<td>0.07141957</td>
<td>1.3335813</td>
<td>0.182</td>
<td>-0.04474169</td>
</tr>
<tr>
<td>b2</td>
<td>0.4205147</td>
<td>0.32383556</td>
<td>1.2985438</td>
<td>0.194</td>
<td>-0.21419136</td>
</tr>
</tbody>
</table>

---

**meta.lm.cor**

*Meta-regression analysis for Pearson or partial correlations*

**Description**

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a Fisher-transformed Pearson or partial correlation. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. The correlations are Fisher-transformed and hence the parameter estimates do not have a simple interpretation. However, the hypothesis test results can be used to decide if a population slope is either positive or negative.

**Usage**

```
meta.lm.cor(alpha, n, cor, s, X)
```
Arguments

alpha  alpha level for 1-alpha confidence
n      vector of sample sizes
cor    vector of estimated Pearson or partial correlations
s      number of control variables
X      matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - Standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

Examples

n <- c(55, 190, 65, 35)
cor <- c(.40, .65, .60, .45)
q <- 0
x1 <- c(18, 25, 23, 19)
X <- matrix(x1, 4, 1)
meta.lm.cor(.05, n, cor, q, X)

# Should return:
# Estimate SE   z  p   LL   UL
# b0 -0.4783215 0.48631509 -0.983563 0.325 -1.431481595 0.47483852
# b1  0.05047154 0.02128496 2.371231 0.018 0.008753794 0.09218929

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a Fisher-transformed correlation. The correlations can be of different types (e.g., Pearson, partial, Spearman). The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. This function uses estimated correlations and their standard errors as input. The correlations are Fisher-transformed and hence the parameter estimates do not have a simple interpretation. However, the hypothesis test results can be used to decide if a population slope is either positive or negative.
meta.lm.cronbach

Usage

meta.lm.cor.gen(alpha, cor, se, X)

Arguments

alpha  alpha level for 1-alpha confidence
cor    vector of estimated correlations
se     number of control variables
X      matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

Examples

cor <- c(.40, .65, .60, .45)
se <- c(.182, .114, .098, .132)
x1 <- c(18, 25, 23, 19)
X <- matrix(x1, 4, 1)
meta.lm.cor.gen(.05, cor, se, X)

# Should return:
# Estimate SE z p
# b0 -0.47832153 0.63427931 -0.7541181 0.451
# b1 0.05047154 0.02879859 1.7525699 0.080

meta.lm.cronbach  Meta-regression analysis for Cronbach reliabilities

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a log-complement Cronbach reliability. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. The exponentiated slope estimate for a predictor variable describes a multiplicative change in variable.
Usage

meta.lm.cronbach(alpha, n, rel, r, X)

Arguments

alpha
alpha level for 1-alpha confidence

n
vector of sample sizes

rel
vector of estimated reliabilities

r
number of measurements (e.g., items)

X
matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - exponentiated OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the exponentiated confidence interval
- UL - upper limit of the exponentiated confidence interval

References


Examples

```r
n <- c(583, 470, 546, 680)
rel <- c(.91, .89, .90, .89)
x1 <- c(1, 0, 0, 0)
X <- matrix(x1, 4, 1)
meta.lm.cronbach(.05, n, rel, 10, X)
# Should return:
# Estimate    SE   z    p    LL    UL
#   b0 -2.2408328 0.03675883 -60.960391 0.000 -2.3128788 -2.16878684
#   b1 -0.1689006 0.07204625 -2.344336 0.019 -0.3101087 -0.02769259
```
**meta.lm.gen**  
*Meta-regression analysis for any type of effect size*

**Description**
This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is any type of effect size. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

**Usage**

```
meta.lm.gen(alpha, est, se, X)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha</td>
<td>alpha level for 1-alpha confidence</td>
</tr>
<tr>
<td>est</td>
<td>vector of parameter estimates</td>
</tr>
<tr>
<td>se</td>
<td>vector of standard errors</td>
</tr>
<tr>
<td>X</td>
<td>matrix of predictor values</td>
</tr>
</tbody>
</table>

**Value**
Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

**Examples**

```r
est <- c(4.1, 4.7, 4.9, 5.7, 6.6, 7.3)
se <- c(1.2, 1.5, 1.3, 1.8, 2.0, 2.6)
x1 <- c(10, 20, 30, 40, 50, 60)
x2 <- c(1, 1, 1, 0, 0, 0)
X <- matrix(cbind(x1, x2), 6, 2)
meta.lm.gen(.05, est, se, X)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>3.5333333</td>
<td>4.374683</td>
<td>0.807678</td>
<td>0.419</td>
<td>-5.04089</td>
<td>12.1075</td>
</tr>
<tr>
<td>b1</td>
<td>0.060000</td>
<td>0.090588</td>
<td>0.6623</td>
<td>0.508</td>
<td>-0.1175</td>
<td>0.2375</td>
</tr>
<tr>
<td>b2</td>
<td>-0.166667</td>
<td>2.811398</td>
<td>-0.0593</td>
<td>0.953</td>
<td>-5.6769</td>
<td>5.3435</td>
</tr>
</tbody>
</table>
meta.lm.mean.ps  

Meta-regression analysis for paired-samples mean differences

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a paired-samples mean difference. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

Usage

meta.lm.mean.ps(alpha, m1, m2, sd1, sd2, cor, n, X)

Arguments

alpha  
alpha level for 1-alpha confidence
m1  
vector of estimated means for group 1
m2  
vector of estimated means for group 2
sd1  
vector of estimated SDs for group 1
sd2  
vector of estimated SDs for group 2
cor  
vector of estimated correlations
n  
vector of sample sizes
X  
matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- t - t-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- df - degrees of freedom

References

Examples

```r
n <- c(65, 30, 29, 45, 50)
cor <- c(.87, .92, .85, .90, .88)
m1 <- c(20.1, 20.5, 19.3, 21.5, 19.4)
m2 <- c(10.4, 10.2, 8.5, 10.3, 7.8)
sd1 <- c(9.3, 9.9, 10.1, 10.5, 9.8)
sd2 <- c(7.8, 8.0, 8.4, 8.1, 8.7)
x1 <- c(2, 3, 3, 4, 4)
X <- matrix(x1, 5, 1)
meta.lm.mean.ps(.05, m1, m2, sd1, sd2, cor, n, X)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>t</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>8.00</td>
<td>1.249</td>
<td>6.40</td>
<td>0.000</td>
<td>5.537</td>
<td>10.46</td>
</tr>
<tr>
<td>b1</td>
<td>0.85</td>
<td>0.379</td>
<td>2.24</td>
<td>0.026</td>
<td>0.108</td>
<td>1.598</td>
</tr>
</tbody>
</table>

meta.lm.mean.ps

Meta-regression analysis for 1-group means

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a mean from one group. The estimates are OLS estimates with standard errors that accomodate residual heteroscedasticity.

Usage

```r
meta.lm.mean.ps(alpha, m, sd, n, X)
```

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `m`: vector of estimated means
- `sd`: vector of estimated standard deviations
- `n`: vector of sample sizes
- `X`: matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- t - t-value
• p - p-value
• LL - lower limit of the confidence interval
• UL - upper limit of the confidence interval
• df - degrees of freedom

Examples

\[
n <- c(25, 15, 30, 25, 40) \\
m <- c(20.1, 20.5, 19.3, 21.5, 19.4) \\
sd <- c(10.4, 10.2, 8.5, 10.3, 7.8) \\
x1 <- c(1, 1, 0, 0, 0) \\
x2 <- c(12, 13, 11, 13, 15) \\
X <- matrix(cbind(x1, x2), 5, 2) \\
meta.lm.mean1(.05, m, sd, n, X)
\]

# Should return:
# | Estimate | SE     | t     | p      | LL    | UL    | df |
# |----------|--------|-------|--------|-------|-------|----|
# | b0       | 19.45490196 | 6.7873381 | 2.8663522 | 0.005 | 6.0288763 | 32.880928 | 132 |
# | b1       | 0.25686275 | 1.9834765 | 0.12950128 | 0.897 | -3.6666499 | 4.180375 | 132 |
# | b2       | 0.04705882 | 0.5064693 | 0.09291544 | 0.926 | -0.9547876 | 1.049095 | 132 |

### meta.lm.mean2

**Meta-regression analysis for 2-group mean differences**

**Description**

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a 2-group mean difference. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

**Usage**

\[
\text{meta.lm.mean2(alpha, m1, m2, sd1, sd2, n1, n2, X)}
\]

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `m1`: vector of estimated means for group 1
- `m2`: vector of estimated means for group 2
- `sd1`: vector of estimated SDs for group 1
- `sd2`: vector of estimated SDs for group 2
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `X`: matrix of predictor values
Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- **Estimate** - OLS estimate
- **SE** - standard error
- **t** - t-value
- **p** - p-value
- **LL** - lower limit of the confidence interval
- **UL** - upper limit of the confidence interval
- **df** - degrees of freedom

References


Examples

```r
n1 <- c(65, 30, 29, 45, 50)
n2 <- c(67, 32, 31, 20, 52)
m1 <- c(31.1, 32.3, 31.9, 29.7, 33.0)
m2 <- c(34.1, 33.2, 30.6, 28.7, 26.5)
sd1 <- c(7.1, 8.1, 7.8, 6.8, 7.6)
sd2 <- c(7.8, 7.3, 7.5, 7.2, 6.8)
x1 <- c(4, 6, 7, 7, 8)
x2 <- c(1, 0, 0, 0, 1)
X <- matrix(cbind(x1, x2), 5, 2)
meta.lm.meanratio.ps(.05, m1, m2, sd1, sd2, n1, n2, X)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>t</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>-15.20</td>
<td>3.4097610</td>
<td>-4.457791</td>
<td>0.000</td>
<td>-21.902415</td>
<td>-8.497585</td>
</tr>
<tr>
<td>b1</td>
<td>2.35</td>
<td>0.4821523</td>
<td>4.873979</td>
<td>0.000</td>
<td>1.402255</td>
<td>3.297745</td>
</tr>
<tr>
<td>b2</td>
<td>2.85</td>
<td>1.5358109</td>
<td>1.855697</td>
<td>0.064</td>
<td>-0.168875</td>
<td>5.868875</td>
</tr>
</tbody>
</table>

---

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a paired-samples log mean ratio. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.
meta.lm.meanratio.ps

Usage

meta.lm.meanratio.ps(alpha, m1, m2, sd1, sd2, cor, n, X)

Arguments

alpha alpha level for 1-alpha confidence
m1 vector of estimated means for group 1
m2 vector of estimated means for group 2
sd1 vector of estimated SDs for group 1
sd2 vector of estimated SDs for group 2
cor vector of estimated correlations
n vector of sample sizes
X matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - the exponentiated estimate
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval

Examples

n <- c(65, 30, 29, 45, 50)
cor <- c(.87, .92, .85, .90, .88)
m1 <- c(20.1, 20.5, 19.3, 21.5, 19.4)
m2 <- c(10.4, 10.2, 8.5, 10.3, 8.1)
sd1 <- c(9.3, 9.9, 10.1, 10.5, 9.8)
sd2 <- c(7.8, 8.0, 8.4, 8.1, 8.7)
x1 <- c(2, 3, 3, 4, 4)
X <- matrix(x1, 5, 1)
meta.lm.meanratio.ps(.05, m1, m2, sd1, sd2, cor, n, X)

# Should return:
# Estimate SE LL UL z p
# b0 0.50957008 0.13000068 0.254773424 0.7643667 3.919749 0.000
# b1 0.07976238 0.04133414 -0.001251047 0.1607758 1.929697 0.054
# exp(Estimate) exp(LL) exp(UL)
meta.lm.meanratio2

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the
dependent variable is a 2-group log mean ratio. The estimates are OLS estimates with standard
errors that accommodate residual heteroscedasticity.

Usage

meta.lm.meanratio2(alpha, m1, m2, sd1, sd2, n1, n2, X)

Arguments

alpha alpha level for 1-alpha confidence
m1 vector of estimated means for group 1
m2 vector of estimated means for group 2
sd1 vector of estimated SDs for group 1
sd2 vector of estimated SDs for group 2
n1 vector of group 1 sample sizes
n2 vector of group 2 sample sizes
X matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix
has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - the exponentiated estimate
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval
Examples

```r
n1 <- c(65, 30, 29, 45, 50)
n2 <- c(67, 32, 31, 20, 52)
m1 <- c(31.1, 32.3, 31.9, 29.7, 33.0)
m2 <- c(34.1, 33.2, 30.6, 28.7, 26.5)
sd1 <- c(7.1, 8.1, 7.8, 6.8, 7.6)
sd2 <- c(7.8, 7.3, 7.5, 7.2, 6.8)
x1 <- c(4, 6, 7, 7, 8)
X <- matrix(x1, 5, 1)
meta.lm.odds(.05, m1, m2, sd1, sd2, n1, n2, X)
```

# Should return:
# Estimate SE LL UL z p
# b0 -0.40208954 0.09321976 -0.58479692 -0.21938216 -4.313351 0
# b1 0.06831545 0.01484125 0.03922712 0.09740377 4.603078 0
# exp(Estimate) exp(LL) exp(UL)
# b0 0.6689208 0.557219 0.8030148
# b1 1.0707030 1.040007 1.1023054
```

---

**meta.lm.odds**

Meta-regression analysis for odds ratios

**Description**

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a log odds ratio. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. The exponentiated slope estimate for a predictor variable describes a multiplicative change in the odds ratio associated with a 1-unit increase in that predictor variable.

**Usage**

```r
meta.lm.odds(alpha, f1, f2, n1, n2, X)
```

**Arguments**

- **alpha**: alpha level for 1-alpha confidence
- **f1**: vector of group 1 event counts
- **f2**: vector of group 2 event counts
- **n1**: vector of group 1 sample sizes
- **n2**: vector of group 2 sample sizes
- **X**: matrix of predictor values
Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- exp(Estimate) - the exponentiated estimate
- exp(LL) - lower limit of the exponentiated confidence interval
- exp(UL) - upper limit of the exponentiated confidence interval

References


Examples

```r
n1 <- c(204, 201, 932, 130, 77)
n2 <- c(106, 103, 415, 132, 83)
f1 <- c(24, 40, 93, 14, 5)
f2 <- c(12, 9, 28, 3, 1)
x1 <- c(4, 4, 5, 3, 26)
x2 <- c(1, 1, 1, 0, 0)
X <- matrix(cbind(x1, x2), 5, 2)
meta.lm.odds(.05, f1, f2, n1, n2, X)
# Should return:
# Estimate SE z p LL UL
# b0 1.541895013 0.69815801 2.20851868 0.027 0.1735305 2.91025958
# b1 -0.004417932 0.04840623 -0.09126784 0.927 -0.0992924 0.09045653
# b2 -1.071122269 0.60582695 -1.76803337 0.077 -2.2585213 0.11627674
# exp(Estimate) exp(LL) exp(UL)
# b0 4.6734381 1.1894969 18.361564
# b1 0.9955918 0.9054779 1.094674
# b2 0.3426238 0.1045049 1.123307
```
meta.lm.prop.ps  Meta-regression analysis for paired-samples proportion differences

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a paired-samples proportion difference. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

Usage

meta.lm.prop.ps(alpha, f11, f12, f21, f22, X)

Arguments

alpha  alpha level for 1-alpha confidence
f11    vector of frequencies in cell 1,1
f12    vector of frequencies in cell 1,2
f21    vector of frequencies in cell 2,1
f22    vector of frequencies in cell 2,2
X      matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References

Examples

```r
f11 <- c(40, 20, 25, 30)
f12 <- c(3, 2, 2, 1)
f21 <- c(7, 6, 8, 6)
f22 <- c(26, 25, 13, 25)
x1 <- c(1, 1, 4, 6)
x2 <- c(1, 1, 0, 0)
X <- matrix(cbind(x1, x2), 4, 2)
meta.lm.prop.ps(.05, f11, f12, f21, f22, X)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>-0.21113402</td>
<td>0.21119823</td>
<td>-0.9996960</td>
<td>-0.62507494</td>
<td>0.20280690</td>
</tr>
<tr>
<td>b1</td>
<td>0.02185567</td>
<td>0.03861947</td>
<td>0.5659236</td>
<td>-0.05383711</td>
<td>0.09754845</td>
</tr>
<tr>
<td>b2</td>
<td>0.12575138</td>
<td>0.17655623</td>
<td>0.7122455</td>
<td>-0.22029248</td>
<td>0.47179524</td>
</tr>
</tbody>
</table>

**meta.lm.prop1**

*Meta-regression analysis for 1-group proportions*

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a proportion from one group. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

Usage

```r
meta.lm.prop1(alpha, f, n, X)
```

Arguments

- **alpha**: alpha level for 1-alpha confidence
- **f**: vector of sample frequency counts
- **n**: vector of sample sizes
- **X**: matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- **Estimate**: OLS estimate
- **SE**: standard error
- **z**: z-value
- **p**: p-value
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval
Examples

```r
f <- c(38, 26, 24, 15, 45, 38)
n <- c(80, 60, 70, 50, 180, 200)
x1 <- c(10, 15, 18, 22, 24, 30)
X <- matrix(x1, 6, 1)
meta.lm.prop1(.05, f, n, X)
```

# Should return:
# Estimate SE z p LL UL
# b0 0.63262816 0.06845707 9.241239 0 0.49845477 0.766801546
# b1 -0.01510565 0.00290210 -5.205076 0 -0.02079367 -0.009417641

---

`meta.lm.prop2`  
*Meta-regression analysis for 2-group proportion differences*

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a 2-group proportion difference. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

Usage

```r
meta.lm.prop2(alpha, f1, f2, n1, n2, X)
```

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `f1`: vector of group 1 event counts
- `f2`: vector of group 2 event counts
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `X`: matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
**meta.lm.propratio2**

**Description**

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a log proportion ratio. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. The exponentiated slope estimate for a predictor variable describes a multiplicative change in the proportion ratio associated with a 1-unit increase in that predictor variable.

**Usage**

```r
meta.lm.propratio2(alpha, f1, f2, n1, n2, X)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `f1`: vector of group 1 event counts
- `f2`: vector of group 2 event counts
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `X`: matrix of predictor values

**Examples**

```r
f1 <- c(24, 40, 93, 14, 5)
f2 <- c(12, 9, 28, 3, 1)
n1 <- c(204, 201, 932, 130, 77)
n2 <- c(106, 103, 415, 132, 83)
x1 <- c(4, 4, 5, 3, 26)
x2 <- c(1, 1, 1, 0, 0)
X <- matrix(cbind(x1, x2), 5, 2)
meta.lm.propratio2(.05, f1, f2, n1, n2, X)
```

# Should return:

```
<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>0.089756283</td>
<td>0.034538077</td>
<td>2.5987632</td>
<td>0.009</td>
<td>0.02206290</td>
</tr>
<tr>
<td>b1</td>
<td>-0.001447968</td>
<td>0.001893097</td>
<td>-0.7648672</td>
<td>0.444</td>
<td>-0.00515837</td>
</tr>
<tr>
<td>b2</td>
<td>-0.034670988</td>
<td>0.034125708</td>
<td>-1.0159786</td>
<td>0.310</td>
<td>-0.10155615</td>
</tr>
</tbody>
</table>
```
**Value**

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- **Estimate** - OLS estimate
- **SE** - standard error
- **z** - z-value
- **p** - p-value
- **LL** - lower limit of the confidence interval
- **UL** - upper limit of the confidence interval
- **exp(Estimate)** - the exponentiated estimate
- **exp(LL)** - lower limit of the exponentiated confidence interval
- **exp(UL)** - upper limit of the exponentiated confidence interval

**References**


**Examples**

```r
n1 <- c(204, 201, 932, 130, 77)
n2 <- c(106, 103, 415, 132, 83)
f1 <- c(24, 40, 93, 14, 5)
f2 <- c(12, 9, 28, 3, 1)
x1 <- c(4, 4, 5, 3, 26)
x2 <- c(1, 1, 1, 0, 0)
X <- matrix(cbind(x1, x2), 5, 2)
meta.lm.propratio2(.05, f1, f2, n1, n2, X)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
<th>exp(Estimate)</th>
<th>exp(LL)</th>
<th>exp(UL)</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>1.4924887636</td>
<td>0.69172794</td>
<td>2.15762393</td>
<td>0.0315</td>
<td>0.13672691</td>
<td>2.84825062</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b1</td>
<td>0.000579509</td>
<td>0.04999884</td>
<td>0.01151928</td>
<td>0.9910</td>
<td>-0.09741998</td>
<td>0.09857188</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b2</td>
<td>-1.0837844594</td>
<td>0.59448206</td>
<td>-1.82307345</td>
<td>0.0676</td>
<td>-2.24894789</td>
<td>0.08137897</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
meta.lm.semipart

Meta-regression analysis for semipartial correlations

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a Fisher-transformed semipartial correlation. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. The correlations are Fisher-transformed and hence the parameter estimates do not have a simple interpretation. However, the hypothesis test results can be used to decide if a population slope is either positive or negative.

Usage

meta.lm.semipart(alpha, n, cor, r2, X)

Arguments

- alpha: alpha level for 1-alpha confidence
- n: vector of sample sizes
- cor: vector of estimated semipartial correlations
- r2: vector of estimated squared multiple correlations for full model
- X: matrix of predictor values

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate: OLS estimate
- SE: standard error
- z: z-value
- p: p-value
- LL: lower limit of the confidence interval
- UL: upper limit of the confidence interval

Examples

```r
n <- c(128, 97, 210, 217)
cor <- c(.35, .41, .44, .39)
r2 <- c(.29, .33, .36, .39)
x1 <- c(18, 25, 23, 19)
X <- matrix(x1, 4, 1)
meta.lm.semipart(.05, n, cor, r2, X)
```

# Should return:
## meta.lm.spear

Meta-regression analysis for Spearman correlations

### Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a Fisher-transformed Spearman correlation. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity. The correlations are Fisher-transformed and hence the parameter estimates do not have a simple interpretation. However, the hypothesis test results can be used to decide if a population slope is either positive or negative.

### Usage

```r
meta.lm.spear(alpha, n, cor, X)
```

### Arguments

- `alpha`: alpha level for 1-alpha confidence
- `n`: vector of sample sizes
- `cor`: vector of estimated Spearman correlations
- `X`: matrix of predictor values

### Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- **Estimate**: OLS estimate
- **SE**: standard error
- **z**: z-value
- **p**: p-value
- **LL**: lower limit of the confidence interval
- **UL**: upper limit of the confidence interval

### Example

```r
# b0 0.1969598 0.3061757 0.6432905 0.520 -0.40313339 0.79705315
# b1 0.01055584 0.0145696 0.7245114 0.469 -0.01800004 0.03911172
```
Examples

n <- c(150, 200, 300, 200, 350)
cor <- c(.14, .29, .16, .21, .23)
x1 <- c(18, 25, 23, 19, 24)
X <- matrix(x1, 5, 1)
meta.lm.spear(.05, n, cor, X)

# Should return:
# Estimate SE z p LL UL
# b0 -0.08920088 0.26686388 -0.3342561 0.738 -0.612244475 0.43384271
# b1  0.01370866 0.01190212 1.1517825 0.249 -0.009619077 0.03703639

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a paired-samples standardized mean difference. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.

Usage

meta.lm.stdmean.ps(alpha, m1, m2, sd1, sd2, cor, n, X, stdzr)

Arguments

alpha alpha level for 1-alpha confidence
m1 vector of estimated means for group 1
m2 vector of estimated means for group 2
sd1 vector of estimated SDs for group 1
sd2 vector of estimated SDs for group 2
cor vector of estimated correlations
n vector of sample sizes
X matrix of predictor values
stdzr
  • set to 0 for square root unweighted average variance standardizer
  • set to 1 for group 1 SD standardizer
  • set to 2 for group 2 SD standardizer
Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- t - t-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

```r
n <- c(65, 30, 29, 45, 50)
cor <- c(.87, .92, .85, .90, .88)
m1 <- c(20.1, 20.5, 19.3, 21.5, 19.4)
m2 <- c(10.4, 10.2, 8.5, 10.3, 7.8)
sd1 <- c(9.3, 9.9, 10.1, 10.5, 9.8)
sd2 <- c(7.8, 8.0, 8.4, 8.1, 8.7)
x1 <- c(2, 3, 3, 4, 4)
X <- matrix(x1, 5, 1)
meta.lm.stdmean.ps(.05, m1, m2, sd1, sd2, cor, n, X, 0)
```

# Should return:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>b0</td>
<td>1.01740253</td>
<td>0.25361725</td>
<td>4.0115667</td>
<td>0.000</td>
<td>0.5203218</td>
</tr>
<tr>
<td>b1</td>
<td>0.04977943</td>
<td>0.07755455</td>
<td>0.6418635</td>
<td>0.521</td>
<td>-0.1022247</td>
</tr>
</tbody>
</table>

---

**meta.lm.stdmean2**

*Meta-regression analysis for 2-group standardized mean differences*

Description

This function estimates the intercept and slope coefficients in a meta-regression model where the dependent variable is a 2-group standardized mean difference. The estimates are OLS estimates with standard errors that accommodate residual heteroscedasticity.
Usage

`meta.lm.stdmean2(alpha, m1, m2, sd1, sd2, n1, n2, X, stdzr)`

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `m1`: vector of estimated means for group 1
- `m2`: vector of estimated means for group 2
- `sd1`: vector of estimated SDs for group 1
- `sd2`: vector of estimated SDs for group 2
- `n1`: vector of group 1 sample sizes
- `n2`: vector of group 2 sample sizes
- `X`: matrix of predictor values
- `stdzr`:
  - set to 0 for square root unweighted average variance standardizer
  - set to 1 for group 1 SD standardizer
  - set to 2 for group 2 SD standardizer
  - set to 3 for square root weighted average variance standardizer

Value

Returns a matrix. The first row is for the intercept with one additional row per predictor. The matrix has the following columns:

- Estimate - OLS estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

```r
n1 <- c(65, 30, 29, 45, 50)
n2 <- c(67, 32, 31, 20, 52)
m1 <- c(31.1, 32.3, 31.9, 29.7, 33.0)
m2 <- c(34.1, 33.2, 30.6, 28.7, 26.5)
sd1 <- c(7.1, 8.1, 7.8, 6.8, 7.6)
sd2 <- c(7.8, 7.3, 7.5, 7.2, 6.8)
x1 <- c(4, 6, 7, 7, 8)
```
X <- matrix(x1, 5, 1)
meta.lm.stdmean2(.05, m1, m2, sd1, sd2, n1, n2, X, 0)

# Should return:
# Estimate    SE    z     p      LL     UL
# b0 -1.6988257 0.4108035 -4.135373 0 -2.5039857 -0.8936657
# b1  0.2871641 0.0649815  4.419167 0  0.1598027  0.4145255

meta.sub.cor

Confidence interval for a difference in average Pearson or partial correlations for two sets of studies

Description
Computes the estimate, standard error, and confidence interval for a difference in average Pearson or partial correlations for two mutually exclusive sets of studies. Each set can have one or more studies. All of the correlations must be either Pearson correlations or partial correlations.

Usage
meta.sub.cor(alpha, n, cor, s, group)

Arguments
alpha      alpha level for 1-alpha confidence
n          vector of sample sizes
cor        vector of estimated Pearson correlations
s          number of control variables (set to 0 for Pearson)
group      vector of group indicators:
  • 1 for set A
  • 2 for set B
  • 0 to ignore

Value
Returns a matrix with three rows:
  • Row 1 - estimate for Set A
  • Row 2 - estimate for Set B
  • Row 3 - estimate for difference, Set A - Set B

The columns are:
  • Estimate - estimate of average correlation or difference
  • SE - standard error
  • LL - lower limit of the confidence interval
  • UL - upper limit of the confidence interval
**References**


**Examples**

```r	n <- c(55, 190, 65, 35)
cor <- c(.40, .65, .60, .45)
group <- c(1, 1, 2, 0)
meta.sub.cor(.05, n, cor, 0, group)

# Should return:
# Estimate SE  LL   UL
# Set A:  0.525 0.06195298 0.3932082 0.6356531
# Set B:  0.600 0.08128008 0.4171458 0.7361686
# Set A - Set B: -0.075 0.10219894 -0.2645019 0.1387283
```

---

**meta.sub.cronbach**  
*Confidence interval for a difference in average Cronbach reliabilities for two sets of studies*

**Description**

Computes the estimate, standard error, and confidence interval for a difference in average Cronbach reliability coefficients for two mutually exclusive sets of studies. Each set can have one or more studies. The number of measurements used to compute the sample reliability coefficient is assumed to be the same for all studies.

**Usage**

```r
meta.sub.cronbach(alpha, n, rel, r, group)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `n`: vector of sample sizes
- `rel`: vector of estimated Cronbach reliabilities
- `r`: number of measurements (e.g., items)
- `group`: vector of group indicators:
  - 1 for set A
  - 2 for set B
  - 0 to ignore
Value

Returns a matrix with three rows:

- Row 1 - estimate for Set A
- Row 2 - estimate for Set B
- Row 3 - estimate for difference, Set A - Set B

The columns are:

- Estimate - estimate of average correlation or difference
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

n <- c(120, 170, 150, 135)
rel <- c(.89, .87, .73, .71)
group <- c(1, 1, 2, 2)
r <- 10
meta.sub.cronbach(.05, n, rel, r, group)

# Should return:
# Estimate   SE    LL    UL
# Set A:    0.88  0.1068845  0.8581268  0.8999386
# Set B:    0.72  0.02515130  0.6684484  0.7668524
# Set A - Set B: 0.16  0.02732821  0.1082933  0.2152731

meta.sub.pbcord

Confidence interval for a difference in average point-biserial correlations for two sets of studies

Description

Computes the estimate, standard error, and confidence interval for a difference in average point-biserial correlations for two mutually exclusive sets of studies. Each set can have one or more studies. Two types of point-biserial correlations can be analyzed. One type uses an unweighted variance and is appropriate for 2-group experimental designs. The other type uses a weighted variance and is appropriate for 2-group nonexperimental designs with simple random sampling. Equal variances within or across studies is not assumed.
Usage

```r
meta.sub.pbcor(alpha, m1, m2, sd1, sd2, n1, n2, type, group)
```

Arguments

- `alpha` alpha level for 1-alpha confidence
- `m1` vector of estimated means for group 1
- `m2` vector of estimated means for group 2
- `sd1` vector of estimated SDs for group 1
- `sd2` vector of estimated SDs for group 2
- `n1` vector of group 1 sample sizes
- `n2` vector of group 2 sample sizes
- `type` set to 1 for weighted variance, set to 2 for unweighted variance
- `group` vector of group indicators:
  - 1 for set A
  - 2 for set B
  - 0 to ignore

Value

Returns a matrix with three rows:

- Row 1 - estimate for Set A
- Row 2 - estimate for Set B
- Row 3 - estimate for difference, Set A - Set B

The columns are:

- Estimate - estimate of average correlation or difference
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References

Examples

```r
m1 <- c(45.1, 39.2, 36.3, 34.5)
m2 <- c(30.0, 35.1, 35.3, 36.2)
sd1 <- c(10.7, 10.5, 9.4, 11.5)
sd2 <- c(12.3, 12.0, 10.4, 9.6)
n1 <- c(40, 20, 50, 25)
n2 <- c(40, 20, 48, 26)
group <- c(1, 1, 2, 2)
meta.sub.pbcor(.05, m1, m2, sd1, sd2, n1, n2, 2, group)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set A</td>
<td>0.36338772</td>
<td>0.08552728</td>
<td>0.1854777</td>
<td>0.5182304</td>
</tr>
<tr>
<td>Set B</td>
<td>-0.01480511</td>
<td>0.08741322</td>
<td>-0.1840491</td>
<td>0.1552914</td>
</tr>
<tr>
<td>Set A - Set B</td>
<td>0.37819284</td>
<td>0.12229467</td>
<td>0.1320530</td>
<td>0.6075828</td>
</tr>
</tbody>
</table>

---

`meta.sub.semipart` *Confidence interval for a difference in average semipartial correlations for two sets of studies*

Description

Computes the estimate, standard error, and confidence interval for a difference in average semipartial correlations for two sets of mutually exclusive studies. Each set can have one or more studies.

Usage

```r
meta.sub.semipart(alpha, n, cor, r2, group)
```

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `n`: vector of sample sizes
- `cor`: vector of estimated semi-partial correlations
- `r2`: vector of squared multiple correlations for a model that includes the IV and all control variables
- `group`: vector of group indicators:
  - 1 for set A
  - 2 for set B
  - 0 to ignore
Value

Returns a matrix with three rows:

- Row 1 - estimate for Set A
- Row 2 - estimate for Set B
- Row 3 - estimate for difference, Set A - Set B

The columns are:

- Estimate - estimate of average correlation or difference
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

Examples

n <- c(55, 190, 65, 35)
cor <- c(.40, .65, .60, .45)
r2 <- c(.25, .41, .43, .39)
group <- c(1, 1, 2, 0)
meta.sub.semipart(.05, n, cor, r2, group)

# Should return:
# Estimate SE LL UL
# Set A: 0.525 0.05955276 0.3986844 0.6317669
# Set B: 0.600 0.07931155 0.4221127 0.7333949
# Set A - Set B: -0.075 0.09918091 -0.2587113 0.1324682

Description

Computes the estimate, standard error, and confidence interval for a difference in average Spearman correlations for two mutually exclusive sets of studies. Each set can have one or more studies.

Usage

meta.sub.spear(alpha, n, cor, group)
Arguments

alpha       alpha level for 1-alpha confidence
n           vector of sample sizes
cor         vector of estimated Spearman correlations
group       vector of group indicators:
            • 1 for set A
            • 2 for set B
            • 0 to ignore

Value

Returns a matrix with three rows:

• Row 1 - estimate for Set A
• Row 2 - estimate for Set B
• Row 3 - estimate for difference, Set A - Set B

The columns are:

• Estimate - estimate of average correlation or difference
• SE - standard error
• LL - lower limit of the confidence interval
• UL - upper limit of the confidence interval

References


Examples

n <- c(55, 190, 65, 35)
cor <- c(.40, .65, .60, .45)
group <- c(1, 1, 2, 0)
meta.sub.spear(.05, n, cor, group)

# Should return:
#   Estimate   SE     LL     UL
# Set A:   0.525 0.06483629 0.3865928 0.6402793
# Set B:   0.600 0.08829277 0.3992493 0.7458512
# Set A - Set B: -0.075 0.10954158 -0.2760700 0.1564955
replicate.cor  Compares Pearson or partial correlations in original and follow-up studies

Description
This function can be used to compare and combine Pearson or partial correlations from the original study and the follow-up study. The confidence level for the difference is 1 – 2\(\alpha\).

Usage
replicate.cor(\(\alpha\), cor1, n1, cor2, n2, \(s\))

Arguments
- \(\alpha\): alpha level for 1-\(\alpha\) confidence
- cor1: estimated Pearson correlation between y and x in original study
- n1: sample size in original study
- cor2: estimated Pearson correlation between y and x in follow-up study
- n2: sample size in follow-up study
- \(s\): number of control variables in each study (0 for Pearson)

Value
A 4-row matrix. The rows are:
- Row 1 summarizes the original study
- Row 2 summarizes the follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size between the two studies

The columns are:
- Estimate - Pearson or partial correlation estimate
- SE - standard error
- \(z\) - \(z\)-value
- \(p\) - \(p\)-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References
Examples

```
replicate.cor(.05, .598, 80, .324, 200, 0)
```

# Should return:

<table>
<thead>
<tr>
<th>#</th>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>Original</td>
<td>0.598</td>
<td>0.11396058</td>
<td>6.589418</td>
<td>4.708045e−09</td>
<td>0.4355043</td>
</tr>
<tr>
<td>#</td>
<td>Follow-up</td>
<td>0.324</td>
<td>0.07124705</td>
<td>4.819037</td>
<td>2.865955e−06</td>
<td>0.1939787</td>
</tr>
<tr>
<td>#</td>
<td>Original - Follow-up</td>
<td>0.274</td>
<td>0.09708614</td>
<td>2.633335</td>
<td>8.455096e−03</td>
<td>0.1065496</td>
</tr>
<tr>
<td>#</td>
<td>Average</td>
<td>0.461</td>
<td>0.04854307</td>
<td>7.634998</td>
<td>2.264855e−14</td>
<td>0.3725367</td>
</tr>
</tbody>
</table>

**Description**

This function can be used to compare and combine any effect size (e.g., odds ratio, proportion ratio, proportion difference, slope coefficient, etc.) using the effect size estimate and its standard error from the original study and the follow-up study. The same results can be obtained using the `meta.lc.gen` function with appropriate contrast coefficients. The confidence level for the difference is 1 − 2α.

**Usage**

```
replicate.gen(alpha, est1, se1, est2, se2)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `est1`: estimated effect size in original study
- `se1`: effect size standard error in original study
- `est2`: estimated effect size in follow-up study
- `se2`: effect size standard error in follow-up study

**Value**

A 4-row matrix. The rows are:

- Row 1 summarizes the original study
- Row 2 summarizes the follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size between the two studies

Columns are:

- Estimate - effect size estimate
• SE - standard error
• z - z-value
• p - p-value
• LL - lower limit of the confidence interval
• UL - upper limit of the confidence interval

References

Examples
replicate.gen(.05, .782, .210, .650, .154)
# Should return:
# Estimate SE z p LL UL
# Original: 0.782 0.2100000 3.7238095 1.962390e-04 0.3704076 1.1935924
# Follow-up: 0.650 0.1540000 4.2207792 2.434593e-05 0.3481655 0.9518345
# Original - Follow-up: 0.132 0.2604151 0.5068831 6.122368e-01 -0.2963446 0.5603446
# Average: 0.716 0.1302075 5.4989141 3.821373e-08 0.4607979 0.9712021

replicate.mean.ps (Compares paired-samples mean differences in original and follow-up studies)

Description
This function computes confidence intervals for a paired-samples mean difference from an original study and a follow-up study. Confidence intervals for the difference and average effect size also are computed. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence intervals for the difference and average. The same results can be obtained using the meta.lc.mean.ps function with appropriate contrast coefficients. The confidence level for the difference is 1 – 2alpha.

Usage
replicate.mean.ps(
   alpha,
   m11,
   m12,
   sd11,
   sd12,
   cor1,
   n1,
Arguments

- alpha: alpha level for 1-alpha confidence
- m11: estimated mean for group 1 in original study
- m12: estimated mean for group 2 in original study
- sd11: estimated SD for group 1 in original study
- sd12: estimated SD for group 2 in original study
- cor1: estimated correlation of paired observations in original study
- n1: sample size in original study
- m21: estimated mean for group 1 in follow-up study
- m22: estimated mean for group 2 in follow-up study
- sd21: estimated SD for group 1 in follow-up study
- sd22: estimated SD for group 2 in follow-up study
- cor2: estimated correlation of paired observations in follow-up study
- n2: sample size in follow-up study

Value

A 4-row matrix. The rows are:

- Row 1 summarizes the original study
- Row 2 summarizes the follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size between the two studies

The columns are:

- Estimate - effect size estimate
- SE - standard error
- df - degrees of freedom
- t - t-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
replicate.mean2

References

Examples
replicate.mean.ps(.05, 86.22, 70.93, 14.89, 12.32, .765, 20, 84.81, 77.24, 15.68, 16.95, .702, 75)

# Should return:
# Estimate SE t p
# Original: 15.29 2.154344 7.097288 9.457592e-07
# Follow-up: 7.57 1.460664 5.182575 1.831197e-06
# Original - Follow-up: 7.72 2.602832 2.966000 5.166213e-03
# Average: 11.43 1.301416 8.782740 1.010232e-10
# LL UL df
# Original: 10.780906 19.79909 19.00000
# Follow-up: 4.659564 10.48044 74.00000
# Original - Follow-up: 3.332885 12.10712 38.40002
# Average: 8.796322 14.06368 38.40002

replicate.mean2 Compares 2-group mean differences in original and follow-up studies

Description
This function computes confidence intervals for a 2-group mean difference from an original study and a follow-up study. Confidence intervals for the difference and average effect size also are computed. A Satterthwaite adjustment to the degrees of freedom is used to improve the accuracy of the confidence intervals. The same results can be obtained using the meta.lc.mean2 function with appropriate contrast coefficients. The confidence level for the difference is 1 – 2alpha.

Usage
replicate.mean2(
alpha,
m11,
m12,
sd11,
sd12,
n11,
n12,
m21,
m22,
sd21,
Arguments

alpha alpha level for 1-alpha confidence
m11 estimated mean for group 1 in original study
m12 estimated mean for group 2 in original study
sd11 estimated SD for group 1 in original study
sd12 estimated SD for group 2 in original study
n11 sample size for group 1 in original study
n12 sample size for group 2 in original study
m21 estimated mean for group 1 in follow-up study
m22 estimated mean for group 2 in follow-up study
sd21 estimated SD for group 1 in follow-up study
sd22 estimated SD for group 2 in follow-up study
n21 sample size for group 1 in follow-up study
n22 sample size for group 2 in follow-up study

Value

A 4-row matrix. The rows are:

- Row 1 summarizes the original study
- Row 2 summarizes the follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size between the two studies

The columns are:

- Estimate - effect size estimate
- SE - standard error
- t - t-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- df - degrees of freedom

References

Examples

replicate.mean2(.05, 21.9, 16.1, 3.82, 3.21, 40, 40,
25.2, 19.1, 3.98, 3.79, 75, 75)

# Should return:
# Estimate SE t p
# Original: 5.80 0.7889312 7.3517180 1.927969e-10
# Follow-up: 6.10 0.6346075 9.6122408 0.000000e+00
# Original - Follow-up: -0.30 1.0124916 -0.2962988 7.673654e-01
# Average: 5.95 0.5062458 11.7531843 0.000000e+00
#    LL   UL  df
# Original: 4.228624 7.371376 75.75255
# Follow-up: 4.845913 7.354087 147.64728
# Original - Follow-up: -1.974571 1.374571 169.16137
# Average: 4.950627 6.949373 169.16137

replicate.oddsratio

Compares 2-group proportion differences in original and follow-up studies

Description

This function computes confidence intervals for an odds ratio from an original study and a follow-up study. Confidence intervals for the ratio and average odds ratio size also are computed. The confidence level for the difference is 1 – 2alpha.

Usage

replicate.oddsratio(alpha, est1, se1, est2, se2)

Arguments

alpha       alpha level for 1-alpha confidence
est1        estimate of log odds ratio 1 in original study
se1         standard error of log odds ration in original study
est2        estimate of log odds ratio in follow-up study
se2         standard error of log odds ratio in follow-up study

Value

A 4-row matrix. The rows are:

- Row 1 summarizes the Original study
- Row 2 summarizes the Follow-up study
- Row 3 estimates the difference between studies
• Row 4 estimates the average effect size of the two studies

The columns are:

• Estimate - effect size estimate
• SE - standard error
• z - z-value
• p - p-value
• LL - exponentiated lower limit of the confidence interval
• UL - exponentiated upper limit of the confidence interval

References


Examples

replicate.oddsratio(.05, 1.39, .302, 1.48, .206)

# Should return:
# Estimate SE z p
# Original: 1.39000000 0.3020000 4.6026490 4.171509e-06
# Follow-up: 1.48000000 0.2060000 7.1844660 6.747936e-13
# Original - Follow-up: -0.06273834 0.3655681 -0.1716188 8.637372e-01
# Average: 0.36067292 0.1827840 1.9732190 4.847061e-02
# exp(LL) exp(UL)
# Original: 2.2212961 7.256583
# Follow-up: 2.9336501 6.578144
# Original - Follow-up: 0.5147653 1.713551
# Average: 1.0024257 2.052222

replicate.plot

*Plot to compare estimates from an original and follow-up study*

Description

Generates a basic plot using ggplot2 to visualize the estimates from and original and follow-up study
replicate.plot

Usage

replicate.plot(
  result,
  focus = c("Both", "Difference", "Average"),
  reference_line = NULL,
  diamond_height = 0.2,
  difference_axis_ticks = 5,
  ggtheme = ggplot2::theme_classic()
)

Arguments

result
  • a result matrix from any of the replicate functions in vcmeta
focus
  • Optional specification of the focus of the plot; defaults to 'Both'
  • Both - a bit busy: plots each estimate, difference, and average
  • Difference - plot each estimate and difference between them
  • Average - plot each estimate and the average effect size
reference_line
  Optional x-value for a reference line. Only applies if focus is 'Difference' or 'Both'. Defaults to NULL, in which case a reference line is not drawn.
diamond_height
  • Optional height of the diamond representing average effect size. Only applies if focus is 'Average' or 'Both'. Defaults to 0.2
difference_axis_ticks
  • Optional requested number of ticks on the difference axis. Only applies if focus is 'Difference' or 'Both'. Defaults to 5.
ggtheme
  • optional ggplot2 theme object; defaults to theme_classic()

Value

Returns a ggplot object. If stored, can be further customized via the ggplot API

Examples

# Compare Damisch et al., 2010 to Calin-Jageman & Caldwell 2014
# Damisch et al., 2010, Exp 1, German participants made 10 mini-golf putts
# Half were told they had a 'lucky' golf ball; half were not
# Found a large but uncertain improvement in shots made in the luck condition
# Calin-Jageman & Caldwell, 2014, Exp 1, was a pre-registered replication with
# input from Damisch, though with English-speaking participants
# Here we compare the effect sizes, in original units, for the two studies
# Use the replicate.mean2 function because the design is 2-group between-subs

library(ggplot2)
damisch_v_calinjageman_raw <- replicate.mean2(
  alpha = 0.05,
  m11 = 6.42,
  m12 = 4.75,
sd11 = 1.88,
sd12 = 2.15,
n11 = 14,
n12 = 14,
m21 = 4.73,
m22 = 4.62,
sd21 = 1.958,
sd22 = 2.12,
n21 = 66,
n22 = 58
)

# View the comparison:
damisch_v_calinjageman_raw

# Now plot the comparison, focusing on the difference
replicate.plot(damisch_v_calinjageman_raw, focus = "Difference")

# Plot the comparison, focusing on the average
replicate.plot(damisch_v_calinjageman_raw,
focus = "Average",
reference_line = 0,
diamond_height = 0.1
)

# Kind of busy, but plot the comparison with both difference and average
# In this case, store the plot for manipulation
myplot <- replicate.plot(
    damisch_v_calinjageman_raw,
    focus = "Both",
    reference_line = 0
)

# View the stored plot
myplot

# Change x-labels and study labels
myplot <- myplot + xlab("Difference in Putts Made, Lucky - Control")
myplot <- myplot + scale_y_discrete(
    labels = c(
        "Average",
        "Difference",
        "Calin-Jageman & Caldwell, 2014",
        "Damisch et al., 2010"
    )
)

# View the updated plot
myplot

Description

This function computes confidence intervals for a 2-group proportion difference from an original study and a follow-up study. Confidence intervals for the difference and average effect size also are computed. The same results can be obtained using the meta.lc.prop2 function with appropriate contrast coefficients. The confidence level for the difference is $1 - 2\alpha$.

Usage

```
replicate.prop2(alpha, f11, f12, n11, n12, f21, f22, n21, n22)
```

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `f11`: sample count for group 1 in original study
- `f12`: sample count for group 2 in original study
- `n11`: sample size for group 1 in original study
- `n12`: sample size for group 2 in original study
- `f21`: sample count for group 1 in follow-up study
- `f22`: sample count for group 2 in follow-up study
- `n21`: sample size for group 1 in follow-up study
- `n22`: sample size for group 2 in follow-up study

Value

A 4-row matrix. The rows are:

- Row 1 summarizes the Original study
- Row 2 summarizes the Follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size of the two studies

The columns are:

- Estimate - effect size estimate
- SE - standard error
- z - z-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
References


Examples

```r
replicate.prop2(.05, 21, 16, 40, 40, 19, 13, 60, 60)
```

# Should return:

<table>
<thead>
<tr>
<th>#</th>
<th>Estimate</th>
<th>SE</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td># Original:</td>
<td>0.11015594</td>
<td>0.06771053</td>
<td>1.6268656</td>
<td>0.1037656</td>
</tr>
<tr>
<td># Follow-up:</td>
<td>0.09273105</td>
<td>0.03308263</td>
<td>2.7878811</td>
<td>0.0052661</td>
</tr>
<tr>
<td># Original - Follow-up:</td>
<td>-0.01742489</td>
<td>-0.03469880</td>
<td>-0.5160220</td>
<td>0.6078097</td>
</tr>
<tr>
<td># Average:</td>
<td>0.10239041</td>
<td>0.06403809</td>
<td>1.5947106</td>
<td>0.1117411</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td># Original:</td>
<td>-0.09273105</td>
<td>0.3308263</td>
</tr>
<tr>
<td># Follow-up:</td>
<td>-0.05933787</td>
<td>0.2528863</td>
</tr>
<tr>
<td># Original - Follow-up:</td>
<td>-0.19915727</td>
<td>0.2463384</td>
</tr>
<tr>
<td># Average:</td>
<td>-0.02255427</td>
<td>0.2428661</td>
</tr>
</tbody>
</table>

---

**replicate.slope**

*Computes confidence intervals for a slope in original and follow-up studies*

Description

Computes confidence intervals for a slope in original and follow-up studies, the difference in slopes, and the average of the slopes (equal error variances between studies is not assumed). The confidence interval for the difference uses a 1 - 2alpha confidence level. Use the replicate.gen function for slopes in other types of models (e.g., binary logistic, ordinal logistic, SEM).

Usage

```r
replicate.slope(alpha, b1, se1, n1, b2, se2, n2, s)
```

Arguments

- `alpha`: alpha level for 1-alpha or 1 - 2alpha confidence
- `b1`: sample slope in original study
- `se1`: standard error of slope in original study
- `n1`: sample size in original study
- `b2`: sample slope in follow-up study
- `se2`: standard error of slope in follow-up study
- `n2`: sample size in follow-up study
- `s`: number of predictor variables in model
Value

A 4-row matrix. The rows are:

- Row 1 summarizes the Original study
- Row 2 summarizes the Follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size of the two studies

The columns are:

- Estimate - effect size estimate
- SE - standard error
- t - t-value
- p - p-value
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval
- df - degrees of freedom

References


Examples

```
replicate.slope(.05, 23.4, 5.16, 50, 18.5, 4.48, 90, 4)
# Should return:
# Estimate    SE       t       p
# Original:  23.40 5.160000 4.5348837 4.250869e-05
# Follow-up: 18.50 4.480000 4.1294643 8.465891e-05
# Original - Follow-up: -4.90 6.833447 0.7170612 4.749075e-01
# Average:  20.95 3.416724 6.1316052 1.504129e-08
#    LL    UL    df
# Original: 13.007227 33.79277 45.0000
# Follow-up: 9.592560 27.40744 85.0000
# Original - Follow-up: -6.438743 16.23874 106.4035
# Average:  14.176310 27.72369 106.4035
```
replicate.stdmean.ps  
*Compares paired-samples standardized mean differences in original and follow-up studies*

**Description**

This function computes confidence intervals for a paired-samples standardized mean difference from an original study and a follow-up study. Confidence intervals for the difference and average effect size also are computed. The same results can be obtained using the `meta.lc.stdmean.ps` function with appropriate contrast coefficients. The confidence level for the difference is $1 - 2\alpha$.

**Usage**

```r
replicate.stdmean.ps(  
  alpha,  
  m11,  
  m12,  
  sd11,  
  sd12,  
  cor1,  
  n1,  
  m21,  
  m22,  
  sd21,  
  sd22,  
  cor2,  
  n2  
)
```

**Arguments**

- `alpha`: alpha level for 1-alpha confidence
- `m11`: estimated mean for group 1 in original study
- `m12`: estimated mean for group 2 in original study
- `sd11`: estimated SD for group 1 in original study
- `sd12`: estimated SD for group 2 in original study
- `cor1`: estimated correlation of paired observations in original study
- `n1`: estimated size in original study
- `m21`: estimated mean for group 1 in follow-up study
- `m22`: estimated mean for group 2 in follow-up study
- `sd21`: estimated SD for group 1 in follow-up study
- `sd22`: estimated SD for group 2 in follow-up study
- `cor2`: estimated correlation of paired observations in follow-up study
- `n2`: sample size in follow-up study
Value

A 4-row matrix. The rows are:

- Row 1 summarizes the original study
- Row 2 summarizes the follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size between the two studies

The columns are:

- Estimate - effect size estimate
- SE - standard error
- LL - lower limit of the confidence interval
- UL - upper limit of the confidence interval

References


Examples

```r
replicate.stdmean.ps(alpha = .05,
m11 = 86.22, m12 = 70.93, sd11 = 14.89, sd12 = 12.32, cor1 = .765, n1 = 20,
m21 = 84.81, m22 = 77.24, sd21 = 15.68, sd22 = 16.95, cor2 = .702, n2 = 75)
```

# Should return:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>SE</th>
<th>LL</th>
<th>UL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orginal</td>
<td>1.0890300</td>
<td>0.22915553</td>
<td>0.6697353</td>
<td>1.5680085</td>
</tr>
<tr>
<td>Follow-up</td>
<td>0.4604958</td>
<td>0.09590586</td>
<td>0.2756687</td>
<td>0.6516096</td>
</tr>
<tr>
<td>Original - Follow-up</td>
<td>0.6552328</td>
<td>0.24841505</td>
<td>0.2466264</td>
<td>1.0638392</td>
</tr>
<tr>
<td>Average</td>
<td>0.7747629</td>
<td>0.12420752</td>
<td>0.5313206</td>
<td>1.0182052</td>
</tr>
</tbody>
</table>

**replicate.stdmean2**

Compares 2-group standardized mean differences in original and follow-up studies

Description

This function computes confidence intervals for a 2-group standardized mean difference from an original study and a follow-up study. Confidence intervals for the difference and average effect size also are computed. The same results can be obtained using the `meta.lc.stdmean2` function with appropriate contrast coefficients. The confidence level for the difference is $1 - 2\alpha$. 
Usage

```r
replicate.stdmean2(
  alpha,
  m11, m12,
  sd11, sd12,
  n11, n12,
  m21, m22,
  sd21, sd22,
  n21, n22
)
```

Arguments

- `alpha`: alpha level for 1-alpha confidence
- `m11`: estimated mean for group 1 in original study
- `m12`: estimated mean for group 2 in original study
- `sd11`: estimated SD for group 1 in original study
- `sd12`: estimated SD for group 2 in original study
- `n11`: sample size for group 1 in original study
- `n12`: sample size for group 2 in original study
- `m21`: estimated mean for group 1 in follow-up study
- `m22`: estimated mean for group 2 in follow-up study
- `sd21`: estimated SD for group 1 in follow-up study
- `sd22`: estimated SD for group 2 in follow-up study
- `n21`: sample size for group 1 in follow-up study
- `n22`: sample size for group 2 in follow-up study

Value

A 4-row matrix. The rows are:

- Row 1 summarizes the original study
- Row 2 summarizes the follow-up study
- Row 3 estimates the difference between studies
- Row 4 estimates the average effect size between the two studies

The columns are:

- Estimate - effect size estimate
• SE - standard error
• LL - lower limit of the confidence interval
• UL - upper limit of the confidence interval

References


Examples

replicate.stdmean2(.05, 21.9, 16.1, 3.82, 3.21, 40, 40, 25.2, 19.1, 3.98, 3.79, 75, 75)

# Should return:
# Estimate SE LL UL
# Original: 1.62803662 0.2594668 1.1353486 2.1524396
# Follow-up: 1.56170447 0.1870576 1.2030461 1.9362986
# Original - Follow-up: 0.07422178 0.3198649 -0.4519092 0.6003527
# Average: 1.59487055 0.1599325 1.2814087 1.9083324

---

**se.cor**

Computes the standard error for a Pearson or partial correlation

Description

This function can be used to compute the standard error of a Pearson or partial correlation using the estimated correlation, sample size, and number of control variables. The effect size estimate and standard error output from this function can be used as input in the meta.ave.gen, meta.lc.gen, and meta.lm.gen functions in applications where a combination of different types of correlations are used in the

Usage

```
se.cor(cor, s, n)
```

Arguments

- `cor` estimated Pearson or partial correlation
- `s` number of control variables (0 for Pearson)
- `n` sample size
Value

Returns a one-row matrix:

- Estimate - estimate of Pearson or partial correlation
- SE - standard error

References


Examples

```r
se.cor(.40, 0, 55)
```

# Should return:
# Estimate   SE
# Correlation: 0.4 0.116487

---

**se.mean.ps**

Computes the standard error for a paired-samples mean difference

Description

This function can be used to compute the standard error of a paired-samples mean difference using the two estimated means, and sample size. The effect size estimate and standard error output from this function can be used as input in the *meta.ave.gen*, *meta.lc.gen*, and *meta.lm.gen* functions in applications where compatible mean differences from a combination of 2-group and paired-samples experiments are used in the meta-analysis.

Usage

```r
se.mean.ps(m1, m2, sd1, sd2, cor, n)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m1</td>
<td>estimated mean for measurement 1</td>
</tr>
<tr>
<td>m2</td>
<td>estimated mean for measurement 2</td>
</tr>
<tr>
<td>sd1</td>
<td>estimated standard deviation for measurement 1</td>
</tr>
<tr>
<td>sd2</td>
<td>estimated standard deviation for measurement 2</td>
</tr>
<tr>
<td>cor</td>
<td>estimated correlation for measurements 1 and 2</td>
</tr>
<tr>
<td>n</td>
<td>sample size</td>
</tr>
</tbody>
</table>
se.mean2

Value

Returns a one-row matrix:

- Estimate - estimate of mean difference
- SE - standard error

References


Examples

```r
se.mean.ps(23.9, 25.1, 1.76, 2.01, .78, 25)
```

# Should return:
#               Estimate   SE
# Mean difference: -1.2 0.2544833

se.mean2  Computes the standard error for a 2-group mean difference

Description

This function can be used to compute the standard error of a 2-group mean difference using the two estimated means, estimated standard deviations, and sample sizes. The effect size estimate and standard error output from this function can be used as input in the `meta.ave.gen`, `meta.ic.gen`, and `meta.lm.gen` functions in applications where compatible mean differences from a combination of 2-group and paired-samples experiments are used in the meta-analysis.

Usage

```r
se.mean2(m1, m2, sd1, sd2, n1, n2)
```

Arguments

- `m1`: estimated mean for group 1
- `m2`: estimated mean for group 2
- `sd1`: estimated standard deviation for group 1
- `sd2`: estimated standard deviation for group 2
- `n1`: group 1 sample size
- `n2`: group 2 sample size
se.meanratio.ps

Value

Returns a one-row matrix:

• Estimate - estimate of mean difference
• SE - standard error

References


Examples

```r
se.mean2(21.9, 16.1, 3.82, 3.21, 40, 40)
```

# Estimate SE
# Mean difference: 5.8 0.7889312

```r
se.meanratio.ps
```

*Computes the standard error for a paired-samples log mean ratio*

Description

This function can be used to compute the standard error of a paired-samples mean ratio using the estimated means, estimated size. The effect size estimate and standard error output from this function can be used as input in the `meta.ave.gen`, `meta.lc.gen`, and `meta.lm.gen` functions in application where compatible mean ratios from a combination of 2-group and paired-samples experiments are used in the meta-analysis.

Usage

```r
se.meanratio.ps(m1, m2, sd1, sd2, cor, n)
```

Arguments

- `m1`: estimated mean for measurement 1
- `m2`: estimated mean for measurement 2
- `sd1`: estimated standard deviation for measurement 1
- `sd2`: estimated standard deviation for measurement 2
- `cor`: estimated correlation for measurements 1 and 2
- `n`: sample size
se.meanratio2

Value

Returns a one-row matrix:

- Estimate - estimate of log mean ratio
- SE - standard error

References


Examples

se.meanratio.ps(21.9, 16.1, 3.82, 3.21, .748, 40)

# Should return:
# Estimate SE
# Log mean ratio: 0.3076674 0.02130161

se.meanratio2

*Computes the standard error for a 2-group log mean ratio*

Description

This function can be used to compute the standard error of a 2-group mean ratio using the two estimated means, estimated standard deviations, and sample sizes. The effect size estimate and standard error output from this function can be used as input in the `meta.ave.gen`, `meta.lc.gen`, and `link[vcmeta]meta.lm.gen` functions in application where compatible mean ratios from a combination of 2-group and paired-samples experiments are used in the meta-analysis.

Usage

se.meanratio2(m1, m2, sd1, sd2, n1, n2)

Arguments

- m1: estimated mean for group 1
- m2: estimated mean for group 2
- sd1: estimated standard deviation for group 1
- sd2: estimated standard deviation for group 2
- n1: group 1 sample size
- n2: group 2 sample size
se.odds

Value

Returns a one-row matrix:

- Estimate - estimate of log mean ratio
- SE - standard error

References


Examples

```r
se.meanratio2(21.9, 16.1, 3.82, 3.21, 40, 40)
```

# Should return:

```r
# EstimatelSE
# Log mean ratio: 0.3076674 0.041886
```

se.odds

*Computes the standard error for a log odds ratio*

Description

This function computes a log odds ratio and its standard error using the frequency counts and sample sizes in a 2-group design. These frequency counts and sample sizes can be obtained from a 2x2 contingency table. This function is useful in a meta-analysis of odds ratios where some studies report the sample odds ratio and its standard error and other studies only report the frequency counts or a 2x2 contingency table. The log odds ratio and standard error output from this function can be used as input in the `meta.ave.gen`, `meta.lc.gen`, and `meta.lm.gen` functions.

Usage

```r
se.odds(f1, n1, f2, n2)
```

Arguments

- `f1` number of participants who have the outcome of interest in group 1
- `n1` group 1 sample size
- `f2` number of participants who have the outcome of interest in group 2
- `n2` group 2 sample size
se.pbcor

Value

Returns a one-row matrix:

- Estimate - estimate of log odds ratio
- SE - standard error

References


Examples

se.odds(36, 50, 21, 50)

# Should return:
# Estimate     SE
# Log odds ratio: 1.239501 0.4204435

se.pbcor Computes the standard error for a point-biserial correlation

Description

The function computes a point-biserial correlation and its standard error for two types of point-biserial correlations in 2-group designs using the estimated means, estimated standard deviations, and samples sizes. One type of point-biserial correlation uses an unweighted average of variances and is appropriate for 2-group experimental designs. The other type of point-biserial correlation uses a weighted average of variances and is appropriate for 2-group nonexperimental designs with simple random sampling. This function is useful in a meta-analysis of compatible point-biserial correlations where some studies used a 2-group experimental design and other studies used a 2-group nonexperimental design. The effect size estimate and standard error output from this function can be used as input in the meta.ave.gen, meta.lc.gen, and meta.lm.gen functions.

Usage

se.pbcor(m1, m2, sd1, sd2, n1, n2, type)

Arguments

m1 estimated mean for group 1
m2 estimated mean for group 2
sd1 estimated standard deviation for group 1
sd2 estimated standard deviation for group 2
se.prop.ps

n1 group 1 sample size
n2 group 2 sample size
type • set to 1 for weighted variance average
       • set to 2 for unweighted variance average

Value

Returns a one-row matrix:

• Estimate - estimate of point-biserial correlation
• SE - standard error

References


Examples

se.pbcor(21.9, 16.1, 3.82, 3.21, 40, 40, 1)

# Should return:
# Estimate SE
# Point-biserial correlation: 0.6349786 0.05981325

se.prop.ps Computes the Bonett-Price estimate and standard error for a paired-samples proportion difference

Description

This function can be used to compute the standard error of a paired-samples proportion difference using the frequency counts from a 2 x 2 contingency table. The effect size estimate and standard error output from this function can be used as input in the meta.ave.gen, meta.lc.gen, and meta.lm.gen functions in applications where compatible proportion differences from a combination of 2-group and paired-samples studies are used in the meta-analysis.

Usage

se.prop.ps(f00, f01, f10, f11)
Arguments

- **f00**: number of participants with \( y = 0 \) and \( x = 0 \)
- **f01**: number of participants with \( y = 0 \) and \( x = 1 \)
- **f10**: number of participants with \( y = 1 \) and \( x = 0 \)
- **f11**: number of participants with \( y = 1 \) and \( x = 1 \)

Value

Returns a one-row matrix:

- **Estimate**: estimate of proportion difference
- **SE**: standard error

References


Examples

```r
se.prop.ps(16, 64, 5, 15)
```

# Should return:

```r
# Estimate SE
# Proportion difference: 0.5784314 0.05953213
```

Description

This function can be used to compute the standard error of a 2-group proportion difference using the two sample proportions and sample sizes. The effect size estimate and standard error output from this function can be used as input in the `meta.ave.gen`, `meta.lc.gen`, and `meta.lm.gen` functions in applications where compatible proportion differences from a combination of 2-group and paired-samples studies are used in the meta-analysis.

Usage

```r
se.prop2(f1, f2, n1, n2)
```
Arguments

- f1: number of participants in group 1 who have the outcome
- f2: number of participants in group 2 who have the outcome
- n1: group 1 sample size
- n2: group 2 sample size

Value

Returns a one-row matrix:

- Estimate - estimate of proportion difference
- SE - standard error

References


Examples

```r
se.prop2(31, 16, 40, 40)
```

# Should return:

```
  Estimate  SE
Proportion difference: 0.3571429 0.1002777
```

Description

This function can be used to compute the standard error of a semipartial correlation using the estimated correlation, sample size, and squared multiple correlation for the full model. The effect size estimate and standard error output from this function can be used as input in the `meta.ave.gen`, `meta.lc.gen`, and `meta.lm.gen` functions in applications where a combination of different types of correlations are used in the meta-analysis.

Usage

```r
se.semipartial(cor, r2, n)
```
**se.slope**

**Arguments**

- **cor**: estimated Pearson correlation
- **sdy**: estimated standard deviation of the response variable
- **sdx**: estimated standard deviation of the predictor variable
- **n**: sample size

**Description**

This function can be used to compute a slope and its standard error for a simple linear regression model using the estimated Pearson correlation and the estimated standard deviations of response and predictor variables. This function is useful in a meta-analysis of slopes of a simple linear regression model where some studies report the Pearson correlation but not the slope.

**Usage**

```r
se.slope(cor, sdy, sdx, n)
```

**Arguments**

- **cor**: estimated Pearson correlation
- **sdy**: estimated standard deviation of the response variable
- **sdx**: estimated standard deviation of the predictor variable
- **n**: sample size

**Examples**

```r
se.semipartial(.40, .25, 60)
```

# Should return:
# Estimate SE
# Semipartial correlation: 0.4 0.1063262
se.spear

Computes the standard error for a Spearman correlation

Description

This function can be used to compute the standard error of a Spearman correlation using the estimated correlation and sample size. The standard error from this function can be used as input in the meta.ave.gen, meta.lc.gen, and meta.lm.gen functions in applications where a combination of different types of correlations are used in the meta-analysis.

Usage

se.spear(cor, n)

Arguments

cor estimated Spearman correlation

n sample size

Value

Returns a one-row matrix:

- Estimate - estimate of Spearman correlation
- SE - standard error

References


Examples

se.slope(.392, 4.54, 2.89, 60)

# Should return:
#          Estimate       SE
# Slope:  0.6158062  0.1897647
se.stdmean.ps

References


Examples

se.spear(.40, 55)

# Should return:
# Estimate  SE
# Spearman correlation:  0.4 0.1210569

se.stdmean.ps Computes the standard error for a paired-samples standardized mean difference

Description

The effect size estimate and standard error output from this function can be used as input in the meta.ave.gen, meta.lc.gen, and meta.lm.gen functions in applications where compatible standardized mean differences from a combination of 2-group and paired-samples experiments are used in the meta-analysis.

Usage

se.stdmean.ps(m1, m2, sd1, sd2, cor, n, stdzr)

Arguments

m1 sample mean for measurement 1
m2 sample mean for measurement 2
sd1 sample standard deviation for measurement 1
sd2 sample standard deviation for measurement 2
cor sample correlation for measurements 1 and 2
n sample size
stdzr

• set to 0 for square root average variance standardizer
• set to 1 for group 1 SD standardizer
• set to 2 for group 2 SD standardizer
se.stdmean2

Value

Returns a one-row matrix:

- Estimate - estimate of standardized mean difference
- SE - standard error

References


Examples

se.stdmean.ps(23.9, 25.1, 1.76, 2.01, .78, 25, 0)

# Should return:
# Estimate SE
# Standardized mean difference: -0.6352097 0.1602852

se.stdmean2(m1, m2, sd1, sd2, n1, n2, stdzr)

Arguments

- m1: sample mean for group 1
- m2: sample mean for group 2
- sd1: sample standard deviation for group 1
- sd2: sample standard deviation for group 2
- n1: group 1 sample size

Description

Use the square root average variance standardizer (stdzr = 0) for 2-group experimental designs. Use the square root weighted variance standardizer (stdzr = 3) for 2-group nonexperimental designs with simple random sampling. The single-group standardizers (stdzr = 1 and stdzr = 2) can be used with either 2-group experimental or nonexperimental designs. The effect size estimate and standard error output from this function can be used as input in the meta.ave.gen, meta.lc.gen, and meta.lm.gen functions in applications where compatible standardized mean differences from a combination of 2-group and paired-samples experiments are used in the meta-analysis.

Usage

se.stdmean2(m1, m2, sd1, sd2, n1, n2, stdzr)
se.stdmean2

group 2 sample size

stdzr
• set to 0 for square root average variance standardizer
• set to 1 for group 1 SD standardizer
• set to 2 for group 2 SD standardizer
• set to 3 for square root weighted variance standardizer

Value

Returns a one-row matrix:

• Estimate - estimate of standardized mean difference
• SE - standard error

References


Examples

se.stdmean2(21.9, 16.1, 3.82, 3.21, 40, 40, 0)

# Should return:
# Estimate SE
# Standardized mean difference: 1.643894 0.2629049
Index

ci.fisher, 4
cor.from.t, 4

meta.ave.agree, 5
meta.ave.cor, 6
meta.ave.cronbach, 7
meta.ave.gen, 8, 99–108, 110–112
meta.ave.gen.cc, 10
meta.ave.gen.rc, 11
meta.ave.mean.ps, 12
meta.ave.mean2, 14
meta.ave.metricratio.ps, 15
meta.ave.metricratio2, 16
meta.ave.odds, 18
meta.ave.path, 19
meta.ave.pbcor, 20
meta.ave.plot, 22
meta.ave.prop.ps, 23
meta.ave.prop, 24
meta.ave.propratio2, 26
meta.ave.semipart, 27
meta.ave.slope, 28
meta.ave.spear, 29
meta.ave.stdmean.ps, 30
meta.ave.stdmean2, 32
meta.ave.var, 33
meta.chitest, 34
meta.lc.agree, 35
meta.lc.gen, 36, 84, 99–108, 110–112
meta.lc.mean.ps, 37, 85
meta.lc.mean1, 38
meta.lc.mean2, 39, 87
meta.lc.metricratio.ps, 41
meta.lc.metricratio2, 42
meta.lc.odds, 43
meta.lc.prop.ps, 45
meta.lc.prop1, 46
meta.lc.prop2, 47, 93
meta.lc.propratio2, 48
meta.lc.stdmean.ps, 49, 96
meta.lc.stdmean2, 51, 97
meta.lm.agree, 52
meta.lm.cor, 53
meta.lm.cor, 54
meta.lm.cronbach, 55
meta.lm.gen, 57, 99–102, 104–108, 110–112
meta.lm.mean.ps, 58
meta.lm.mean1, 59
meta.lm.mean2, 60
meta.lm.metricratio.ps, 61
meta.lm.metricratio2, 63
meta.lm.odds, 64
meta.lm.prop.ps, 66
meta.lm.prop1, 67
meta.lm.prop2, 68
meta.lm.propratio2, 69
meta.lm.semipart, 71
meta.lm.spear, 72
meta.lm.stdmean.ps, 73
meta.lm.stdmean2, 74
meta.sub.cor, 76
meta.sub.cronbach, 77
meta.sub.pbcor, 78
meta.sub.semipart, 80
meta.sub.spear, 81
replicate.cor, 83
replicate.gen, 84
replicate.mean.ps, 85
replicate.mean2, 87
replicate.oddsratio, 89
replicate.plot, 90
replicate.prop2, 93
replicate.slope, 94
replicate.stdmean.ps, 96
replicate.stdmean2, 97

se.cor, 99
se.mean.ps, 100
se.mean2, 101
INDEX

se.meanratio.ps, 102
se.meanratio2, 103
se.odds, 104
se.pbcor, 105
se.prop.ps, 106
se.prop2, 107
se.semipartial, 108
se.slope, 109
se.spear, 110
se.stdmean.ps, 111
se.stdmean2, 112