Package 'metaumbrella'

February 28, 2025

Type Package

Title Umbrella Review Package for R

Version 1.1.0

Date 2025-02-28

Author Corentin J Gosling [aut, cre],
Aleix Solanes [aut],
Paolo Fusar-Poli [aut],
Joaquim Radua [aut]

Maintainer Corentin J Gosling <cgosling@parisnanterre.fr>

Description A comprehensive range of facilities to perform umbrella reviews with stratification of the evidence in R. The package accomplishes this aim by building on three core functions that: (i) automatically perform all required calculations in an umbrella review (including but not limited to meta-analyses), (ii) stratify evidence according to various classification criteria, and (iii) generate a visual representation of the results. Note that if you are not familiar with R, the core features of this package are available from a web browser (<https://www.metaumbrella.org/>).

License GPL-3

Imports meta, metaConvert, pwr, powerSurvEpi, readxl, tcltk, withr, writexl, xtable

Encoding UTF-8

LazyData true

Suggests rmarkdown, knitr, testthat (>= 3.0.0), spelling, DT, epiR, esc, metafor

Config/testthat/edition 3

Depends R (>= 2.10)

RoxygenNote 7.2.3

VignetteBuilder knitr

Language en-US

NeedsCompilation no

Repository CRAN

Date/Publication 2025-02-28 11:40:02 UTC

Contents

`metaumbrella-package` *metaumbrella: An Umbrella Review Package for R*

Description

The **metaumbrella** package offers several facilities to assist in data analysis when performing an umbrella review. This package is built around three core functions which automatically perform the statistical analyses required for an umbrella review (the `umbrella()` function), stratify the evidence according to various classification criteria (the `add.evidence()` function) and generate a graphical presentation of the results (the `forest()` function).

- The `umbrella()` function automatically performs meta-analyses and additional calculations needed for an umbrella review. It outputs an object of class "umbrella". The advantage of this function over standard R packages only designed for fitting a single meta-analysis lies, for example, in the possibility of automatically fitting several meta-analyses when input information differs, automatically extracting the necessary information to stratify the evidence,

and automatically performing the additional tests needed (a test for excess significance, a test for publication bias and a jackknife leave-one-out analysis).

- The `add.evidence()` function stratifies the evidence generated by the `umbrella()` function according to a set of pre-specified criteria (those proposed by Prof. Ioannidis or an algorithmic version of GRADE classification), or according to a personalized classification that the users may specify manually. This feature allows users to rely on already developed criteria or to develop new ones that match the specific needs of their umbrella review.
- The `forest()` function creates graphical representations of the results of an umbrella review, including a forest plot along with information on the stratification of evidence.

Well-formatted dataset

One of the specificities of the **metaumbrella** package is that all the functions of this package do not have an argument to specify the name of the variables contained in the dataset of the users. Therefore, it is necessary that the datasets that are passed to the different functions of the package respect a very precise formatting (which we will refer to as `well-formatted dataset`). We present here the rules that must be respected when creating a well-formatted dataset.

The datasets passed to the functions of the **metaumbrella** package should contain information on each individual study pooled in the different meta-analyses included in the umbrella review. The information about each individual study must allow for replication of the meta-analyses. It is therefore necessary that the information contained in a well-formatted dataset allows for estimating the effect size and variance of all individual studies. Ten types of effect size measures are accepted:

- **"SMD"**: standardized mean difference (i.e., Cohen's d)
- **"G"**: Hedges' g
- **"MD"**: mean difference
- **"SMC"**: standardized mean change
- **"R"**: Pearson's correlation
- **"Z"**: Fisher's z
- **"OR"** or **"logOR"**: odds ratio or its logarithm
- **"RR"** or **"logRR"**: risk ratio or its logarithm
- **"HR"** or **"logHR"**: hazard ratio or its logarithm
- **"IRR"** or **"logIRR"**: incidence rate ratio or its logarithm

To estimate the effect size and the variance of each individual study, the **metaumbrella** package allows for flexible inputs. We detail below (A) the variables that are mandatory and must be indicated in a well-formatted dataset, (B) the variables that vary depending on the effect size measure and (C) the variables that are optional but that can be indicated to benefit from certain features of the package. Note that the package includes examples of well-formatted datasets for each effect size measure (`df.SMD`, `df.SMC`, `df.R`, `df.OR`, `df.RR`, `df.HR` and `df.IRR`).

A. Mandatory variables:

The following variables must be included in the dataset regardless of the effect size measure used. The name of these variables (in bold) cannot be changed.

- **meta_review**: a character variable that contains an identifier for the sources of the meta-analyses included in an umbrella review. Typically, this variable contains the name of the first-author of the included meta-analyses.
- **factor**: a character variable that contains an identifier for the risk factors or the interventions whose effect are studied. Importantly, all rows in the dataset with the same `factor` value will be pooled together in a meta-analysis.
- **author** and **year**: character variables identifying the name and the year of publication of each individual study that is included in a meta-analysis. For a given factor, all rows with the same author and year values will be identified as having some type of dependence (see below).
- **measure**: a character variable describing the type of effect size measure used to quantify the effect of the factor and it must be either "SMD", "MD", "G", "SMC", "R", "Z", "OR", "logOR", "RR", "logRR", "HR", "logHR", "IRR" or "logIRR". Note here that if a study reports the numbers of cases and controls in exposed and non-exposed groups but does not report an effect size value (i.e., the value of an OR or RR), we recommend specifying "OR" for case-control studies while "RR" for cohort studies.

B. Required information depending on the effect size measure:

Depending on the effect size measure used, different information must be provided to replicate the meta-analyses. To allow users adapting to the data available in the original articles, several combinations of information can be provided for a given effect size measure. We detail the information that can provided in the dataset to replicate the meta-analyses and we provide several summary tables displaying the various combinations of minimum information required to replicate the meta-analyses.

- **value**: Value of the effect size for each individual study.
- **ci_lo**: Lower bound of the 95% confidence interval around the effect size for each individual study.
- **ci_up**: Upper bound of the 95% confidence interval around the effect size for each individual study.
- **n_sample**: Total number of participants in each individual study.
- **n_cases**: Number of cases in each individual study.
- **n_controls**: Number of controls in each individual study.
- **n_exp**: Number of exposed participants in each individual study.
- **n_nexp**: Number of non-exposed participants in each individual study.
- **n_cases_exp**: Number of cases in the exposed group in each individual study.
- **n_controls_exp**: Number of controls in the exposed group in each individual study.
- **n_cases_nexp**: Number of cases in the non-exposed group in each individual study.
- **n_controls_nexp**: Number of controls in the non-exposed group in each individual study.
- **mean_pre_cases**: Mean of the cases at baseline for each individual study.
- **mean_pre_controls**: Mean of the controls at baseline for each individual study.
- **sd_pre_cases**: Standard deviation of the cases at baseline for each individual study.
- **sd_pre_controls**: Standard deviation of the controls at baseline for each individual study.
- **pre_post_cor**: Correlation between the pre-test and post-test scores (across groups) for each individual study.
- **mean_cases**: Mean of the cases (at follow up) for each individual study.
- **mean_controls**: Mean of the controls (at follow up) for each individual study.

- **sd_cases**: Standard deviation of the cases (at follow up) for each individual study.
- **sd_controls**: Standard deviation of the controls (at follow up) for each individual study.
- **time**: Sum of the person-time of disease-free observation in the exposed and non-exposed groups for each individual study.
- **time_exp**: Person-time of disease-free observation in the exposed group for each individual study.
- **time_nexp**: Person-time of disease-free observation in the non-exposed group for each individual study.

We now present the summary tables indicating the minimum combination of information that should be provided for each individual study to run the analyses. The symbol `X` indicates that the information is provided in a dataset. The symbol + between two information indicates that the two information are mandatory. The symbol | between two information indicates that only one of the two information is required. For each effect size measure, users must provide information on at least one row of the table corresponding to the effect size measure used. Note that users can provide different combination of information for a same factor (e.g., it is possible to include the SMD value + 95% CI + sample sizes for a study and the means/SDs + sample sizes for another study within the same factor). We consider the sample size as mandatory given this information is often critical, but basic meta-analytic calculations will run without them in most cases.

1. "SMD":

mean_cases + mean_controls + sd_cases + sd_controls	n_cases + n_controls	value	se | var	ci_lo + ci_up
X	X	-	-	-
-	X	X	-	-
-	X	X	X	-
-	X	X	-	X

2. "G":

n_cases + n_controls	value	se | var	ci_lo + ci_up
X	X	-	-
X	X	X	-
X	X	-	X

3. "MD":

n_cases + n_controls	value	se | var	ci_lo + ci_up
X	X	X	-
X	X	-	X

4. "SMC":

mean_pre_cases + mean_pre_controls +

sd_pre_cases + sd_pre_controls + mean_cases + mean_controls + sd_cases + sd_controls + pre_post_cor	n_cases + n_controls	value	se | var	ci_lo + ci_up
X	X	-	-	-
-	X	X	X	-
-	X	X	-	X

mean_change_cases + mean_change_controls + sd_change_cases + sd_change_controls	n_cases + n_controls
X	X

***5. "R"**:*

n_sample	value	se | var	ci_lo + ci_up
X	X	-	-
X	X	X	-
X	X	-	X

***6. "Z"**:*

n_sample	value	se | var	ci_lo + ci_up
X	X	-	-
X	X	X	-
X	X	-	X

***7. "OR"** or **"logOR"**:*

n_cases_exp + n_controls_exp + n_cases_nexp + n_controls_nexp	n_exp + n_nexp	n_cases + n_controls	value	se | var	ci_lo + ci_up
X	-	-	-	-	-
-	-	X	X	-	-
-	-	X	X	X	-
-	-	X	X	-	X
-	X	-	X	X	-
-	X	-	X	-	X

8. "RR" or "logRR":

n_cases_exp + n_controls_exp + n_cases_nexp + n_controls_nexp	n_cases + n_controls	value	se | var	ci_lo + ci_up
X	-	-	-	-
-	X	X	X	-
-	X	X	-	X

9. "HR" or "logHR":

n_cases + n_controls	value	se | var	ci_lo + ci_up
X	X	X	-
X	X	-	X

10. "IRR" or "logIRR":

n_cases_exp + n_cases_nexp + time_exp + time_nexp	n_cases	time	value	se | var	ci_lo + ci_up
X	-	-	-	-	-
-	X	X	X	X	-
-	X	X	X	-	X

C. Optional variables:

The following variables do not have to be included in a well-formatted dataset but they can be added to benefit from certain features of the functions. The name of these variables (in bold) cannot be changed.

- **multiple_es**: Reason for the presence of several effect sizes for a unique study (i.e., a study with the same author and year values within the same factor). It must be either "groups" or "outcomes". An example of a well-formatted dataset with multiple outcomes/groups can be found here (`df.OR.multi`) and an example of analysis of a dataset with dependent effect sizes is available in a vignette of the package.
  - `groups`: When "groups" is indicated, it is assumed that the multiple effect sizes for a unique study come from independent subgroups. A unique effect size per study is calculated using the Borenstein's (2009) approach. For each study, the sample size is obtained by summing up all participants from the different groups.
  - `outcomes`: When "outcomes" is indicated, it is assumed that the multiple effect sizes come from multiple outcomes (or time-points) measured within the same sample. Again, a unique effect size per study is calculated using the Borenstein's (2009) approach. Strength of the correlation between the outcomes (or time-points) can be indicated using either the `r` column in your dataset (see below) or the `r` argument of the `umbrella()` function. Indicating the strength of the correlation between the outcomes of a study in the `r` column allows to use different values depending on the study. In contrast, using the `r` argument of `umbrella()` function allows to conveniently set a unique correlation for all studies that do not have any value in the `r` column. For each study, the sample size is obtained by taking the largest sample size for one outcome/time-point.

- **r**: When a study reports multiple effect sizes coming from the measurement of several outcomes (or measurements of the same outcome at different time-points) in the same participants, the `r` column can be used to indicate the value of the correlation coefficient between the effect sizes of a given study. The r value should be (i) within the (-1, 1) range, (ii) constant within a study, and (iii) set as `NA` for studies which do not include multiple effect sizes coming from different outcomes/time-points.
- **pre_post_cor**: The value of the correlation coefficient between baseline and follow-up scores in pre-post studies. You should indicate the mean pre-post correlation across groups. Only needed when using the SMC measure.
- **reverse_es**: Whether users want to reverse the effect size of a study. All rows with a `"reverse"` value in this column will have the direction of their effect size flipped (e.g., an OR of 0.5 will be expressed as 2). Note that the reverse_es column has an action on both the direction of the value of an effect size and on the information used to calculate an effect size (e.g., if the means and SDs of experimental and control groups are reported, the mean and SD of the experimental group are used as the mean and SD of the control group and vice-versa). This feature is particularly useful to facilitate the presentation of the results when several meta-analyses report the same effects in opposite direction.
- **rob**: The overall risk of bias of each individual study. Should be either "high", "low" or "unclear". These values are used to generate the "GRADE" classification and to stratify evidence according to the 'rob' criteria in the 'Personalized' classification. Studies with a missing rob are assumed to be at high risk of bias. The approach used to provide a categorical judgment ("low" vs. "unclear" vs. "high) on the risk of bias of a study is left to the user.
- **rob1_rand, rob1_allocation, rob1_blind_pers, rob1_blind_outcome, rob1_attrition, rob1_report, rob2_rand, rob2_deviation, rob2_missing, rob2_outcome, rob2_report**: All individual risk of bias. Note that the reporting bias (rob1_report & rob2_report) is required to the GRADE stratification.
- **indirectness**: Whether a given factor is at risk of indirectness (only used for the GRADE stratification). Note that the indirectness score ('very serious', 'serious', 'no indirectness') should be constant for a given factor.
- **amstar**: The amstar score of the meta-analysis. Note that the amstar score should be constant for a given factor. These values are used only to stratify evidence according to the 'amstar' criteria in the 'Personalized' classification.
- **analysis**: Whether users want to conduct specific analyses. For now, only the `"allelic"` value can be specified, which multiplies by two the number of cases and controls.
- **discard**: Whether a particular row should be removed from the analyses (any row with a "yes" or TRUE value in the `discard` column will be removed).

`add.evidence` *Add evidence classes to "umbrella" objects*

Description

Add evidence classes to the factors included in an umbrella review.

Usage

```
add.evidence(
  x,
  criteria = "Ioannidis",
  eq_range_or = c(0.8, 1.25),
  eq_range_g = c(-0.1, 0.1),
  class_I = c(n_studies = NA, total_n = NA, n_cases = NA, p_value = NA, I2 = NA,
   imprecision = NA, rob = NA, amstar = NA, egger_p = NA, esb_p = NA, JK_p = NA, pi =
    NA, largest_CI = NA),
  class_II = c(n_studies = NA, total_n = NA, n_cases = NA, p_value = NA, I2 = NA,
   imprecision = NA, rob = NA, amstar = NA, egger_p = NA, esb_p = NA, JK_p = NA, pi =
    NA, largest_CI = NA),
  class_III = c(n_studies = NA, total_n = NA, n_cases = NA, p_value = NA, I2 = NA,
   imprecision = NA, rob = NA, amstar = NA, egger_p = NA, esb_p = NA, JK_p = NA, pi =
    NA, largest_CI = NA),
  class_IV = c(n_studies = NA, total_n = NA, n_cases = NA, p_value = NA, I2 = NA,
   imprecision = NA, rob = NA, amstar = NA, egger_p = NA, esb_p = NA, JK_p = NA, pi =
    NA, largest_CI = NA),
  verbose = TRUE
)
```

Arguments

x	an object of class “umbrella”.
criteria	the evidence criteria. It must be "GRADE", "Ioannidis" or "Personalized".
eq_range_or	a vector of the bounds of equivalence ranges for OR/RR/HR/IRR (only required for GRADE) criteria.
eq_range_g	a vector of the bounds of equivalence ranges for SMD (only required for GRADE) criteria.
class_I	a vector or list of threshold values required for reaching Class I in the Personalized criteria (see details below).
class_II	a vector or list of threshold values required for reaching Class II in the Personalized criteria (see details below).
class_III	a vector or list of threshold values required for reaching Class III in the Personalized criteria (see details below).
class_IV	a vector or list of threshold values required for reaching Class IV in the Personalized criteria (see details below).
verbose	logical variable indicating whether text outputs and messages should be generated. We recommend turning this option to FALSE only after having carefully read all the generated messages.

Details

The `add.evidence()` function performs a stratification of evidence according to three criteria.

"Ioannidis" classification:

This classification allows to stratify evidence according to the criteria described in Fusar-Poli & Radua (2018). This classification proposes to stratify evidence in five ordinal classes: "Class I", "Class II", "Class III", "Class IV", "Class ns". The criteria for each class are the following:

- **Class I:** number of cases > 1000, p-value of the meta-analysis < 10^-6, I^2 < 0.5, 95% prediction interval excluding the null, p-value of the Egger test > .05 and p-value of the excess of statistical significance test > .05.
- **Class II:** number of cases > 1000, p-value of the meta-analysis < 10^-6, largest study with a statistically significant effect and class I criteria not met.
- **Class III:** number of cases > 1000, p-value of the meta-analysis < 10^-3 and class I-II criteria not met.
- **Class IV:** p-value of the meta-analysis < 0.05 and class I-III criteria not met.
- **Class ns:** p-value of the meta-analysis >= 0.05. To apply this classification with R and Z effect size measures, you should indicate both the 'n_sample' AND the 'n_cases'.

"GRADE" classification:

This classification allows to stratify evidence according to four ordinal classes: "High", "Moderate", "Low", "Very low". Importantly, this classification should not be taken as an equivalent to the subjective approach underlying the standard GRADE classification. However, in line with the standard GRADE approach, this classification uses a downgrading procedure in which all factors start with a "High" evidence class that could then be downgraded according to the 5 following criteria. Importantly, when the number of studies is low (k < 5), then the meta-analysis starts with a 'Moderate' rating to account for the difficulty of identifying heterogeneity and publication bias with such a limited number of studies.

All calculations are made automatically, but users should input in their dataset, i) the overall risk of bias of each study ('rob' column), ii) the risk of selective reporting ('rob1_report', or 'rob2_report' columns) and iii) the risk of indirectness ('indirectness' column). If this information is left empty, each criterion will be assumed to be at high risk.

- **Risk of Bias (Limitations):**
  - No downgrade: >=75% of participants included in low-risk studies
  - One downgrade: 50%-75% of participants included in low-risk studies
  - Two downgrades: <=50% of participants included in low-risk studies

  The pooled percentage of participants is calculated as a weighted mean, with weights attributed to each study being equal to the weight each study receives in the meta-analysis.
- **Heterogeneity:**
  - Two downgrades: Substantial discrepancy between the 95% CI and 95% PI (e.g., bounds of the 95% CI and 95% PI not of the same sign and in different equivalence ranges).
  - One downgrade: Small/moderate discrepancy between the 95% CI and 95% PI (e.g., bounds of the 95% CI and 95% PI of the same sign, but in different equivalence ranges).

  When 95% PI is not reliably estimable, the assessment relies on the I² statistic and the percentage of studies with contradicting results:
  - Two downgrades: I² >= 50% and >=10% of studies with statistically significant results in the opposite direction compared to the pooled effect size
  - One downgrade: I² >= 30% and >=10% of studies with statistically significant results in the opposite direction compared to the pooled effect size

- **Indirectness:** The number of downgrades and the criteria are left to the user's discretion, as these factors vary significantly depending on the scope of the review. Examples of criteria may include heterogeneity in participants' age or undefined control groups:
  - No downgrade: No concerns regarding indirectness
  - One downgrade: Serious concerns (e.g., "serious" indirectness)
  - Two downgrades: Very serious concerns (e.g., "very serious" indirectness)
- **Imprecision:**
  - Two downgrades: The 95% CI of the pooled effect size includes both null (SMD = 0; RR/OR = 1) and large (SMD >= 0.80; OR/RR >= 5) effects AND the meta-analysis does not have the sample size required to detect small effects (eSMD = 0.20) with 80% statistical power (n < 394 per arm)
  - One downgrade: The 95% CI of the pooled effect size includes both null and large effects
  - Two downgrades: The meta-analysis does not have the sample size required to detect moderate effects (eSMD = 0.50) with 80% statistical power (n < 64 per arm)
  - One downgrade: The meta-analysis does not have the sample size required to detect small effects (eSMD = 0.20) with 80% statistical power (n < 394 per arm)
- **Publication Bias:**
  - One downgrade: p-value of Egger's test < 0.10, OR excess significance bias p-value < 0.10, OR more than 50% of participants included in trials with high reporting bias

This classification is not available for R and Z effect size measures.

The GRADE classification implementation in this package was developed through the collaborative efforts of: Dr Corentin J. Gosling, Dr Miguel Garcia-Argibay, Prof Richard Delorme, Prof Marco Solmi, Prof Andrea Cipriani, Prof Christoph U. Correll, Dr Cinzia Del Giovane, Prof Paolo Fusar-Poli, Prof Henrik Larsson, Edoardo Ostinelli, Prof Jae Il Shin, Prof DongKeon Yon, Prof Joaquim Radua, Prof John P. Ioannidis and Prof Samuele Cortese

"Personalized" classification:

Because the "Ioannidis" and "GRADE" classifications do not necessarily provide a rating system that perfectly matches the requirements of your umbrella review, the `add.evidence()` function offers the possibility to use a "Personalized" criteria to stratify the evidence according to 13 criteria. This Personalized criteria proposes to stratify the evidence in 5 ordinal classes: "Class I", "Class II", "Class III", "Class IV" and "Class V". "Class I" is the highest class that could be achieved and "Class V" is the lowest.

The overall class achieved by a factor is equal to the lowest class achieved by all the criteria used to stratify evidence. For example, if users choose to stratify the evidence according to 3 criteria (the p-value of the meta-analysis, the inconsistency, the publication bias), and that the classes achieved by these 3 criteria are respectively "Class I", "Class III" and "Class IV", the overall class reached by the factor will be "Class IV".

To determine the class that should be assigned to a factor, users have to indicate - for each class - a vector/list of threshold values for all the criteria that are used to stratify the evidence. A description of the criteria and their corresponding inputs is provided below:

1. `n_studies`: a number of studies included in the meta-analysis. If the number of studies included in the meta-analysis is **strictly superior** to the threshold value indicated in `studies`, the class for which this value is indicated can be reached.
2. `total_n`: a total number of participants included in the meta-analysis. If the total number of participants included in the meta-analysis is **strictly superior** to the threshold value indicated in `total_n`, the class for which this value is indicated can be reached.

3. `n_cases`: a number of cases included in the meta-analysis. If the number of cases included in the meta-analysis is **strictly superior** to the threshold value indicated in `cases`, the class for which this value is indicated can be reached.
4. `p_value`: a p-value of the pooled effect size under the random-effects model. If the p-value of the pooled effect size is **strictly inferior** to the threshold value indicated in `p.value`, the class for which this value is indicated can be reached.
5. `I2`: an i-squared (I^2) value. If the I^2 value of the meta-analysis is **strictly inferior** to the threshold value indicated in `I2`, the class for which this value is indicated can be reached.
6. `imprecision`: a SMD value that will be used to calculate the statistical power of the meta-analysis. If the number of participants included in the meta-analyses allows to obtain a statistical power **strictly superior** to 80% for the SMD value indicated in `imprecision`, the class for which this value is indicated can be reached.
7. `rob`: a percentage of participants included in studies **at low risk of bias**. Note that the approach to determining whether a study is at low risk of bias is left to the user. If the percentage of participants included in studies at low risk of bias is **strictly superior** to the threshold value indicated in `rob`, the class for which this value is indicated can be reached.
8. `amstar`: an AMSTAR rating on the methodological quality of the meta-analysis. If the AMSTAR value of the meta-analysis is **strictly superior** to the threshold value indicated in `amstar`, the class for which this value is indicated can be reached.
9. `egger_p`: a p-value of an Egger's test for publication bias. If the p-value of the Egger's test is **strictly superior** to the threshold value indicated in `egger_p`, the class for which this value is indicated can be reached.
10. `esb_p`: a p-value of a test for excess of statistical significance bias (ESB). If the p-value of the test is **strictly superior** to the threshold value indicated in `esb_p`, the class for which this value is indicated can be reached.
11. `JK_p`: the largest p-value obtained in the jackknife meta-analysis (JK). If the largest p-value obtained in the jackknife meta-analysis is **strictly inferior** to the threshold value indicated in `JK_p`, the class for which this value is indicated can be reached.
12. `pi`: a "notnull" value indicates that users request the 95% prediction interval of the meta-analysis to exclude the null value to achieve the class for which it is indicated.
13. `largest_CI`: a "notnull" value indicates that users request the 95% confidence interval of the largest study included in the meta-analysis to exclude the null value to achieve the class for which it is indicated.

Value

Return an object of class "umbrella" with the evidence classes added.

References

Fusar-Poli, P., & Radua, J. (2018). Ten simple rules for conducting umbrella reviews. *Evidence-Based Mental Health*, **21**, 95-100.

See Also

`umbrella()` for conducting an umbrella review.

Examples

```
### perform calculations required for an umbrella review
df <- subset(df.SMD, factor == "Surgical")
umb.full <- umbrella(df)

### stratify evidence according to the Ioannidis classification
evid_ioannidis <- add.evidence(umb.full, criteria = "Ioannidis")
summary(evid_ioannidis)

### stratify evidence according to the Personalized classification with
### the number of studies and cases, the inconsistency as criteria.
### - a class I can be reached if the number of studies is > 10, the number of cases is > 500 and
###   the I2 is < 25%.
### - a class II can be reached if the number of studies is > 5, the number of cases is > 400 and
###   the I2 is < 50%.
### - a class III can be reached if the number of cases is > 300 and the I2 is < 75%.
### - a class IV can be reached if the number of cases is > 100.
### - else, if the number of cases is <= 100, a class V is assigned.
evid_perso1 <- add.evidence(umb.full, criteria = "Personalized",
   class_I = c(n_studies = 10, n_cases = 500, I2 = 25),
   class_II = c(n_studies = 5, n_cases = 400, I2 = 50),
   class_III = c(n_cases = 300, I2 = 75),
   class_IV = c(n_cases = 100))
summary(evid_perso1)
```

`df.HR` *Meta-analyses exploring the efficacy of several interventions on a binary outcome.*

Description

Fictitious dataset of four meta-analyses of RCTs assessing the efficacy of yoga, aerobic training, resistance training and mindfulness on a binary outcome

Usage

```
df.HR
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the intervention studied.
author	first study author of the individual studies.

year	year of publication of the individual studies.
measure	type of effect size (HR).
value	HR value.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases.
n_controls	number of controls.

Source

No source, the data are entirely fictitious

`df.IRR` *Meta-analysis exploring adverse events of smoking.*

Description

Fictitious dataset of a meta-analysis of prospective cohorts assessing adverse effects of smoking on one binary outcome.

Usage

```
df.IRR
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the factor (only one factor is included).
author	first study author of the individual studies
year	year of publication of the individual studies.
measure	type of effect size (IRR).
value	IRR value.

ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases (sum of the number of cases in the exposed and non-exposed groups).
n_cases_exp	number of cases in the exposed group.
n_cases_nexp	number of cases in the non-exposed group.
time	total person-time at risk (sum of the person-time at risk in the exposed and non-exposed groups).
time_exp	person-time at risk in the exposed group.
time_nexp	person-time at risk in the non-exposed group.

Source

No source, the data are entirely fictitious

`df.MC` *Meta-analyses exploring the efficacy of an intervention on a continuous outcome measured before and after the intervention.*

Description

Fictitious dataset of three meta-analyses of RCTs assessing the efficacy of an intervention on a continuous outcome in 3 populations.

Usage

```
df.MC
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the population studied.
author	first study author of the individual studies.
year	year of publication of the individual studies.

measure	type of effect size (MC).
value	MC value.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases.
n_controls	number of controls.
mean_change_cases	mean change score of patients in the experimental arm.
mean_change_controls	mean change score of patients in the control arm.
sd_change_cases	standard deviations of the change score of patients in the experimental arm.
sd_change_controls	standard deviations of the change score of patients in the control arm.

Source

No source, the data are entirely fictitious

`df.MD` *Meta-analyses exploring the efficacy of surgical and pharmacological interventions.*

Description

Fictitious dataset of two meta-analyses of RCTs assessing the efficacy of surgical and pharmacological interventions on a numeric outcome.

Usage

```
df.MD
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the intervention studied.

author	first study author of the individual studies.
year	year of publication of the individual studies.
measure	type of effect size (MD).
value	MD value.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
mean_cases	means of patients in the experimental arm.
mean_controls	means of patients in the control arm.
sd_cases	standard deviations of patients in the experimental arm.
sd_controls	standard deviations of patients in the control arm.
n_cases	number of patients in the experimental arm.
n_controls	number of patients in the control arm.
rob	risk of bias of the individual studies.
amstar	AMSTAR score of the meta-analysis.

Source

No source, the data are entirely fictitious

`df.OR` *Meta-analyses exploring a risk factor for neurodevelopmental disorders.*

Description

Fictitious dataset of four meta-analyses of cross-sectional studies assessing a risk factor for neurodevelopmental disorders.

Usage

```
df.OR
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the neurodevelopmental disorders on which the effect of the risk factor is studied
author	first study author of the individual studies.
year	year of publication of the individual studies.
measure	type of effect size (OR).
value	OR value.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases (sum of the number of cases in the exposed and non-exposed groups).
n_controls	number of controls (sum of the number of controls in the exposed and non-exposed groups).
n_exp	number of participants in the exposed group (sum of the number of cases and controls in the exposed group).
n_nexp	number of participants in the non-exposed group (sum of the number of cases and controls in the non-exposed group).
n_cases_exp	number of cases in the exposed group.
n_controls_exp	number of controls in the exposed group.
n_cases_nexp	number of cases in the non-exposed group.
n_controls_nexp	number of controls in the non-exposed group.

Source

No source, the data are entirely fictitious

`df.OR.multi` *Meta-analysis of RCTs assessing different dietary interventions on a binary outcome.*

Description

Fictitious dataset including meta-analyses with dependent effect sizes.

Usage

```
df.OR.multi
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the intervention studied.
author	first study author of the individual studies
year	year of publication of the individual studies.
measure	type of effect size (OR).
value	OR value.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases (sum of the number of cases in the exposed and non-exposed groups).
n_controls	number of controls (sum of the number of controls in the exposed and non-exposed groups).
n_cases_exp	number of cases in the exposed group.
n_controls_exp	number of controls in the exposed group.
n_cases_nexp	number of cases in the non-exposed group.
n_controls_nexp	number of controls in the non-exposed group.
multiple_es	indicates the reason of the presence for multiple effect sizes

(due to multiple groups or outcomes) per study.

Source

No source, the data are entirely fictitious

`df.R` *Meta-analyses of correlational data*

Description

Fictitious dataset of four meta-analyses of cross-sectional studies exploring the association between pre- or peri-pregnancy indicators and a numeric variable.

Usage

```
df.R
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the factors.
author	first study author of the individual studies.
year	year of publication of the individual studies.
measure	type of effect size (R).
value	R value.
n_sample	total number of individuals in the sample.

Source

No source, the data are entirely fictitious

`df.radua2019` *Meta-analyses exploring the risk factors for posttraumatic stress disorder.*

Description

Real dataset taken from Tortella-Feliu et al. (2019).

Usage

```
df.radua2019
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the risk factor.
author	first study author of the individual studies.
year	year of publication of the individual studies.
multiple_es	indicates the reason of the presence of multiple effect sizes (due to multiple groups or outcomes) per study.
measure	type of effect size.
value	value of the effect size.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases.
n_controls	number of controls.
n_exp	number of participants in the exposed group (sum of the number of cases and controls in the exposed group).
n_nexp	number of participants in the non-exposed group (sum of the number of cases and controls in the non-exposed group).
n_cases_exp	number of cases in the exposed group.

n_controls_exp	number of controls in the exposed group.
n_cases_nexp	number of cases in the non-exposed group.
n_controls_nexp	number of controls in the non-exposed group.
mean_cases	means of participants in the experimental arm.
sd_cases	standard deviation of participants in the experimental arm.
mean_controls	means of participants in the control arm.
sd_controls	standard deviation of participants in the control arm.
amstar	AMSTAR score of the meta-analysis

Source

Tortella-Feliu, M. and Fullana, M.A., Perez-Vigil, A., Torres, X., Chamorro, J., and Littarelli, S.A., ..., & Radua, J. (2019). Risk Factors for Posttraumatic Stress Disorder: An Umbrella Review of Systematic Reviews and Meta-Analyses. *Neuroscience & Biobehavioral Reviews*, **107**, 154–165.

---

df.RR *Meta-analysis of the adverse events of antidepressants.*

Description

Fictitious dataset of a meta-analysis of cohort studies assessing the risks of adverse outcomes when taking selective serotonin reuptake inhibitors (SSRIs) therapy.

Usage

```
df.RR
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the type of antidepressant studied.
author	first study author of the individual studies.
year	year of publication of the individual studies.

measure type of effect size (RR).

value RR value.

ci_lo lower bound of the 95% confidence interval.

ci_up upper bound of the 95% confidence interval.

n_cases_exp number of cases in the exposed group.

n_exp number of participants in the exposed group (sum of the number of cases and controls in the exposed group).

n_cases_nexp number of cases in the non-exposed group.

n_nexp number of participants in the non-exposed group (sum of the number of cases and controls in the non-exposed group).

Source

No source, the data are entirely fictitious

`df.SMC` *Meta-analyses exploring the efficacy of an intervention on a continuous outcome measured before and after the intervention.*

Description

Fictitious dataset of three meta-analyses of RCTs assessing the efficacy of an intervention on a continuous outcome in 3 populations.

Usage

```
df.SMC
```

Format

The dataset contains the following variables:

meta_review name of the first author of the meta-analysis.

factor name of the population studied.

author first study author of the individual studies.

year year of publication of the individual studies.

measure	type of effect size (SMC).
value	SMC value.
se	standard error of the SMC.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
n_cases	number of cases.
n_controls	number of controls.
mean_cases	means at post-test of patients in the experimental arm.
mean_controls	means at post-test of patients in the control arm.
sd_cases	standard deviations at post-test of patients in the experimental arm.
sd_controls	standard deviations at post-test of patients in the control arm.
mean_pre_cases	means at baseline of patients in the experimental arm.
mean_pre_controls	means at baseline of patients in the control arm.
sd_pre_cases	standard deviations at baseline of patients in the experimental arm.
sd_pre_controls	standard deviations at baseline of patients in the control arm.

Source

No source, the data are entirely fictitious

`df.SMD` *Meta-analyses exploring the efficacy of surgical and pharmacological interventions.*

Description

Fictitious dataset of two meta-analyses of RCTs assessing the efficacy of surgical and pharmacological interventions on a numeric outcome.

Usage

```
df.SMD
```

Format

The dataset contains the following variables:

meta_review	name of the first author of the meta-analysis.
factor	name of the intervention studied.
author	first study author of the individual studies.
year	year of publication of the individual studies.
measure	type of effect size (SMD).
value	SMD value.
se	standard error of the SMD.
ci_lo	lower bound of the 95% confidence interval.
ci_up	upper bound of the 95% confidence interval.
mean_cases	means of patients in the experimental arm.
mean_controls	means of patients in the control arm.
sd_cases	standard deviations of patients in the experimental arm.
sd_controls	standard deviations of patients in the control arm.
n_cases	number of patients in the experimental arm.
n_controls	number of patients in the control arm.
rob	risk of bias of the individual studies.
amstar	AMSTAR score of the meta-analysis.

Source

No source, the data are entirely fictitious

`df.train` *Training dataset*

Description

This is a non-formatted dataset that is used in a vignette to illustrate how obtaining a well-formatted dataset with the help of the `view.errors.umbrella()` function.

Usage

```
df.train
```

Format

The dataset contains the following variables:

comment	comments on studies.
risk_factor	name of the intervention studied.
author_study	first study author of the individual studies.
year_publication_study	year of publication of the individual studies.
type_of_effect_size	type of effect size.
number_of_cases_exposed	number of cases in the exposed group.
number_of_cases_non_exposed	number of cases in the non-exposed group.
number_of_controls_exposed	number of controls in the exposed group.
number_of_controls_non_exposed	number of controls in the non-exposed group.
number_of_participants_exposed	total number of participants in the exposed group.
number_of_participants_non_exposed	total number of participants in the non-exposed group.
number_of_cases	number of cases.
number_of_controls	number of controls.
effect_size_value	value of the effect size
low_bound_ci	lower bound of the 95% confidence interval.
up_bound_ci	upper bound of the 95% confidence interval.
time_disease_free	total person-time at risk (sum of the person-time at risk in the exposed and non-exposed groups).
mean_of_intervention_group	mean of the intervention group

mean_of_control_group	mean of the control group
sd_of_intervention_group	sd of the intervention group
sd_of_control_group	sd of the control group

Source

No source, the data are entirely fictitious

`drop.evidence` *Remove evidence classes from an object of class "umbrella"*

Description

This function removes evidence classes previously created from an object of class "umbrella"

Usage

```
drop.evidence(x)
```

Arguments

`x`	an object of class "umbrella"

Value

Return an object of class "umbrella" with the evidence classes dropped.

References

Fusar-Poli, P., & Radua, J. (2018). Ten simple rules for conducting umbrella reviews. *Evidence-Based Mental Health*, **21**, 95-100.

See Also

`umbrella()` for conducting an umbrella review.

Examples

```
### perform calculations required for an umbrella review
umb.full <- umbrella(df.SMD)

### stratify evidence according to the algorithmic GRADE criteria
evid_grade <- add.evidence(umb.full, criteria = "GRADE")
is.na(evid_grade$Pharmacological$evidence)
```

```
evid_empty <- drop.evidence(evid_grade)
is.na(evid_empty$Pharmacological$evidence)
```

`esb.test` *Perform some tests for excess of significance*

Description

The `esb.test()` function performs various tests to examine the presence of an excess of statistical significance in a given set of studies. These tests aims to determine whether there is an excess in the observed number of studies with statistically significant results compared to what could have been expected.

Usage

```
esb.test(
  x,
  input = "dataframe",
  n_cases = NULL,
  n_controls = NULL,
  measure = NULL,
  method.esb = "TESSPSST",
  true_effect = "UWLS",
  seed = NA,
  tau2 = NULL
)
```

Arguments

`x`	a well-formatted dataset or an object of class "rma" or "meta". If a well-formatted dataset is used, only one factor should be included.
`input`	the type of object used as input. It must be `"dataframe"`, `"rma"` or `"meta"`.
`n_cases`	vector with the number of cases of each included studies. Only required when `x` is an object of class "rma" or "meta". This information can be indicated via the `n_cases` argument of the `esb.test()` function or directly when calling the `rma()` or `meta()` functions (see examples below).
`n_controls`	vector with the number of controls of each included studies. Only required when `x` is an object of class "rma" or "meta" This information can be indicated via the `n_controls` argument of the `esb.test()` function or directly when calling the `rma()` or `meta()` functions (see examples below).
`measure`	the measure of the effect: "SMD", "MD", "R", "Z", "G", "OR" or "logOR, "RR" or "logRR", "HR" or "logHR", "IRR" or "logIRR". If a an object of class "rma" or "meta" is used, the effect size should be either "SMD" or "OR". However, note that for "rma" objects, a SMD is systematically assumed to be a G (to respect the naming used in the **metafor** package). For "meta" objects, a SMD is assumed to be a G unless it is explicitly stated that this is not the case (i.e.,

	using the method.smd = "Cohen" argument). The effect size measure used can be indicated via the measure argument of the esb.test() function or directly when calling the rma() or meta() functions (see examples below).
method.esb	the method used to conduct the test. It must be PSST, TESS or TESSPSST (see details). Default is "TESSPSST".
true_effect	the best approximation of the true effect. It must be "largest", "UWLS" or a numeric value (see details). Default is "UWLS".
seed	an integer value used as an argument by the set.seed() function. Only used for measures "OR", "logOR, "RR", "logRR", "IRR" or "logIRR".
tau2	The tau2 value that should be used when using one of the PSST, TESS or TESSPSST methods (see details).

Details

The function starts by calculating whether each individual study has significant results ($p < .05$). Then, it estimates the statistical power of each individual study to detect an effect size equal to the best approximation of the true effect. The true_effect argument can be used to select the method that will be applied to estimate the true effect.

- If "largest" is entered, the true effect size is assumed to be equal to the effect size of the largest study included in the meta-analysis.
- If "UWLS" is entered, the true effect size is assumed to be equal to unrestricted weighted least squares weighted average.
- If a numeric value is entered, the true effect size is assumed to be equal to the value entered by the user (note that the value of ratios must be in their natural scale).

Last, this function performs a statistical test to determine whether the observed number of statistically significant studies is higher than expected given the mean statistical power. The method.esb argument can be used to select the test.

- If "TESS" is entered, the test of excess statistical significance (TESS) described by Stanley and colleagues (2021) is performed. This test assesses whether the proportion of excess statistical significance is larger than 5%. In this test, power calculations take into account between-study heterogeneity.
- If "PSST" is entered, the proportion of statistical significance test (PSST) described by Stanley and colleagues (2021) is performed. This is a test assessing whether the proportion of statistically significant studies is higher than what could have been expected given the mean statistical power. In this test, power calculations take into account between-study heterogeneity.
- If "TESSPSST" is entered, the function combines results of both "PSST" and "TESS" analyses. "TESSPSST" assumes an excess of statistical significance if at least one of "TESS" and "PSST" is statistically significant.

Value

The dataset contains the following columns:

method	method used to conduct the test.
p.value	p-value for the test statistic.
power	the power of each individual study to detect the best approximation of the true effect (true_effect) at an alpha of .05.
mean_power	the mean power of all individual studies to detect the best approximation of the true effect (true_effect) at an alpha of .05.
k	the total number of studies.
sig	whether each individual study has statistically significant results.
O	the total number of studies with statistically significant results.
E	the total expected number of studies with statistically significant results.

References

Ioannidis, JPA., Munafo, MR., Fusar-Poli, P., Nosek, BA., & David, SP. (2014). Publication and other reporting biases in cognitive sciences: detection, prevalence, and prevention. *Trends in Cognitive Sciences*, **18**, 235-241.

Examples

```
### load a well-formatted dataframe with a single factor
df <- df.SMD[df.SMD$factor == "Surgical", ]

### perform an excess significance bias directly on this dataframe
esb <- esb.test(df, measure = "SMD", input = "dataframe",
                method.esb = "TESS", true_effect = "largest")

### perform an excess significance bias using the umbrella function
esb.umbrella <- umbrella(df, method.esb = "TESS", true_effect = "largest")[[1]]$esb

### perform an excess significance bias on a rma object
### we convert the SMD into Hedges' g
G <- metaumbrella:::.estimate_g_from_d(df$value, df$n_cases, df$n_controls)
rma <- metafor::rma(yi = G$value, sei = G$se,
                    measure = "SMD",
                    ni = df$n_cases + df$n_controls,
                    data = df)

esb.rma <- esb.test(rma, n_cases = df$n_cases, input = "rma", method.esb = "TESS")

### perform an excess significance bias on a meta object
meta <- meta::metagen(TE = G$value, seTE = G$se,
                      sm = "SMD",
```

```
                      n.e = n_cases, n.c = n_controls,
                      data = df)

esb.meta <- esb.test(meta, input = "meta", method.esb = "TESS")

all.equal(esb$p.value, esb.umbrella$p.value, esb.rma$p.value, esb.meta$p.value)
```

`forest` *Forest plots for objects of class "umbrella" or "data.frame"*

Description

Draw a forest plot of the factors included in an umbrella review.

Arguments

`x`	an object of class "umbrella" or "data.frame"
`...`	additional arguments that can be passed to this function

Details

For now, this function simply applies the `forest.umbrella()` function.

Value

In addition to the plot, the function returns a list including:

- a dataset with the factors, their class and their effect size. Particularly useful when adding a column via the 'add_columns' argument to obtain the ordering of the factors in the plot. See the vignette dedicated to the forest plots for a concrete example.
- the optimal width and height of the plot, useful when calling the function `pdf()` or `png()`.

See Also

`forest.umbrella()`

Examples

```
forest(umbrella(df.SMD))
```

`forest.umbrella` *Forest plots for "umbrella" objects*

Description

Draw a forest plot of the factors included in an umbrella review. This function is now a wrapper around the amazing forest.meta function designed by Guido Schwarzer and Gerta Rucker. You can thus add any other arguments available in the forest.meta function but not detailed here.

Usage

```
## S3 method for class 'umbrella'
forest(
  x,
  layout = "meta",
  measure = "auto",
  leftcols = NULL,
  leftlabs = NULL,
  rightcols = NULL,
  rightlabs = NULL,
  digits = 2,
  smlab = "",
  xlab = NULL,
  type.study = "square",
  print.classes = FALSE,
  subgroup = NULL,
  subgroup.name = "",
  col.square = "gray",
  col.study = "black",
  col.square.lines = "black",
  fontsize = 12,
  spacing = 1,
  squaresize = 0.8/spacing,
  ...
)
```

Arguments

`x`	an object of class "umbrella" or a or "data.frame" object returned by the `umbrella()` or `add.evidence()` functions.
`layout`	type of layout of the plot ("meta", "JAMA" or "RevMan5").
`measure`	type of effect size used in the plot ("eG", "eOR", "raw" or "auto"). Default is the most frequently used effect size measure.
`leftcols`	vector of columns contained in the object passed to the `x` argument, used to specify columns which are printed on the left side of the forest plot.

leftlabs	vector of column names used to specify column names printed on the left side of the forest plot.
rightcols	vector of columns contained in the object passed to the x argument, used to specify columns which are printed on the right side of the forest plot.
rightlabs	vector of column names used to specify column names printed on the left side of the forest plot.
digits	number of digits to display
smlab	character string describing the title of the plot
xlab	character string describing the x-axis title
type.study	the shape used to depict the pooled effect size (must be either "square", "diamond", or "predict")
print.classes	a vector of classes. Only factors reaching these classes will be displayed on the plot. These classes must be "I", "II", "III", "IV" and/or "ns" for the "Ioannidis" classification, or "High", "Moderate", "Weak" and/or "Very weak" for the "GRADE" classification, or "I", "II", "III", "IV", and/or "V" for the "Personalized" classification
subgroup	a character variable indicating the name of the column that should be used as header for creating subgroups between the factors
subgroup.name	a character variable displayed just before each modality of the subgroup variable
col.square	The colour for squares reflecting study's weight in the meta-analysis.
col.study	The colour for individual study results and confidence limits.
col.square.lines	The colour for the outer lines of squares reflecting study weight in the meta-analysis.
fontsize	The size of text (in points)
spacing	A numeric variable determining line spacing in a forest plot.
squaresize	A numeric variable used to increase or decrease the size of squares in the forest plot.
...	additional arguments that can be passed to the forest.umbrella function

Details

The function allows to have a visualization of the results of an umbrella review. Various parameters, such as the type of effect size displayed, the restriction to some classes or the color of the dots, allows to simplify the visualization.

Value

Return a forest plot of the pooled effect sizes, along with additional information

References

Balduzzi S, Rucker G, Schwarzer G (2019). How to perform a meta-analysis with R: a practical tutorial. *Evidence-Based Mental Health*, 153–160.

Examples

```
### perform an umbrella review
umb <- umbrella(df.SMD)

### generate a forest plot of each factor included in the umbrella review
forest(umb)

## Not run:
forest(umbrella(df.SMD))

## End(Not run)
```

`overlap.prim` *Overlap in primary studies across reviews*

Description

This function allows to estimate the primary study overlap across reviews

Usage

```
overlap.prim(
  x,
  ID = "factor",
  presentation = "%",
  cut_off = c(0.05, 0.1, 0.15),
  enhanced = TRUE
)
```

Arguments

`x`	an 'umbrella' object
`ID`	a character variable indicating whether the overlap should be looked across 'factor' or across 'meta_review'.
`presentation`	the type of presentation for the overlap. Can be precise percentages ('%') or symbols ('+').
`cut_off`	a vector of 3 cut-off values used only if the 'presentation' argument is set as '+'.
`enhanced`	a logical variable indicating whether you want a narrative description of the information stored in the rows/columns returned by this function.

Value

This function returns a dataframe with 1 row/column per factor or meta-review. As in a correlation matrix, the cell at the intersection of a row and a column contains the desired information (i.e. the overlap between primary studies for the column and row names).

References

Pérez-Bracchiglione, J., Meza, N., Bangdiwala, S. I., Niño de Guzmán, E., Urrútia, G., Bonfill, X., & Madrid, E. (2022). Graphical Representation of Overlap for OVErviews: GROOVE tool. Research synthesis methods, 13(3), 381–388. https://doi.org/10.1002/jrsm.1557

Examples

```
df.SMD$author[22:32] <- df.SMD$author[1:11]
df.SMD$year[22:32] <- df.SMD$year[1:11]
overlap.prim(umbrella(df.SMD),
             presentation = "+", cut_off = c(.05,.15,.25))
```

`print.umbrella` *Print a summary of an object of class "umbrella"*

Description

Print a summary of an object of class "umbrella"

Usage

```
## S3 method for class 'umbrella'
print(x, ...)
```

Arguments

`x`	an object of class "umbrella"
`...`	other arguments that can be passed to the function

Details

Summary method for objects of class "umbrella".

Value

Implicitly calls the `summary.umbrella()` function and displays error or warning messages below the object returned. This is useful when many factors are included in the review and that the results of the `summary.umbrella()` are not stored in an object.

See Also

`summary.umbrella()`

Examples

```
### print the results of an object of class umbrella
umbrella(df.OR.multi, mult.level = TRUE)
```

subset.umbrella *Create a subset of an object of class "umbrella"*

Description

Create a subset of an object of class "umbrella"

Usage

```
## S3 method for class 'umbrella'
subset(x, subset, ...)
```

Arguments

x	an object of class "umbrella".
subset	logical expression indicating elements or rows to keep: missing values are taken as FALSE.
...	other arguments that can be passed to the function

Value

Return an object of class "umbrella" with the results of some factors removed.

Examples

```
### perform calculations required for an umbrella review
umb <- umbrella(df.SMD)

### subset the results to factors "ADHD" & "dyslexia"
subset.umb <- subset.umbrella(umb, unique(df.SMD$factor) %in% c("Pharmacological"))

summary(subset.umb)
```

summary.umbrella *Synthesize information of an object of class "umbrella" in a dataframe*

Description

Synthesize information of an object of class "umbrella" in a dataframe

Usage

```
## S3 method for class 'umbrella'
summary(object, digits = 3, raw_md = FALSE, het_max = FALSE, ...)
```

Arguments

object	an object of class "umbrella"
digits	an integer value specifying the number of decimal places for the rounding of numeric values. Default is 3.
raw_md	a logical variable indicating whether the pooled MD (instead of SMD) should be returned in value, and confidence interval. Do not affect other results and calculations.
het_max	a logical variable indicating whether additional information on heterogeneity should be printed (tau^2, Q-statistic estimate and p-value).
...	other arguments that can be passed to the function

Details

Summary method for objects of class "umbrella".

Value

All main results of an object of class "umbrella" are synthesized in a dataframe, with the results of each factors stored in their own row. Depending on the classification used, the dataframe returned include certain information presented below:

Factor	the name of the factor.
Class	the class assigned during the stratification of evidence.
n_studies	the number of independent studies included in the factor.
total_n	the total number of participants included in the factor.
n_cases	the number of cases included in the factor.
n_controls	the number of controls included in the factor.
measure	the measured used in the calculations.
value	the value of the pooled effect size expressed in its original metric. Note that if a factor includes only one study, its effect size is used as the pooled effect size.
value_CI	the 95% confidence interval (CI) around the pooled effect size expressed in its original metric. Note that if a factor includes only one study, its 95% CI is used as the pooled 95% CI.
eG	the value of the pooled effect size expressed in equivalent Hedges' g (eG).
eG_CI	the 95% CI around the pooled effect size expressed in eG.
eOR	the value of the pooled effect size expressed in equivalent Odds ratio (eOR).

eOR_CI	the 95% CI around the pooled effect size expressed in eOR.
p_value	the p-value of the pooled effect size.
I2	the inconsistency (I^2) value (calculated only if the number of studies in the meta-analysis is equal or larger to 2).
PI_eG	the 95% prediction interval (PI) expressed in eG (calculated only if the number of studies in the meta-analysis is equal or larger to 3).
PI_eOR	the 95% PI expressed in eOR (calculated only if the number of studies in the meta-analysis is equal or larger to 3).
PI_sign	whether the 95% PI includes the null value ("notnull" vs. "null").
egger_p	the p-value of the Egger's test for publication bias (calculated only if the number of studies in the meta-analysis is equal or larger to 3).
egger_sign	whether the p-value of the Egger's test is < .05 ("sig." vs. "ns").
ESB_p	the p-value of the test for excess of significance bias.
ESB_sign	whether the p-value of the excess of significance test is < .05 ("sig." vs. "ns").
power_med	the power to detect a SMD of 0.5 at an alpha of .05 based on the number of cases and controls included in the meta-analysis (when IRR is used as effect size measure, the number of cases and controls in this calculation is assumed to be equal to half the total number of cases included in the meta-analysis).
power	present only in the 'Personalized' classification. - If the user did not use the 'power' criteria to stratify the evidence, this column contains the power to detect a small effect size (SMD = 0.2), a moderate effect size (SMD = 0.5) and a large effect size (SMD = 0.8) at an alpha of .05 based on the number of cases and controls included in the meta-analysis. - If the user used the 'power' criteria to stratify the evidence, this column contains the power to detect the values entered by the user at an alpha of .05 based on the number of cases and controls included in the meta-analysis.
JK_p	the largest p-value obtained in the jackknife leave-one-out meta-analysis (calculated only if the number of studies in the meta-analysis is equal or larger to 2)
JK_sign	whether the largest p-value in the jackknife meta-analysis is < .05 ("sig." vs. "ns")
largest_CI_eG	the 95% CI of the largest study expressed in eG
largest_CI_eOR	the 95% CI of the largest study expressed in eOR

largest_sign	whether the 95% CI of the largest study includes the null value ("notnull" vs. "null")
rob	the percentage of participants included in studies at low risk of bias (calculated only if this information is indicated in the dataset)
amstar	the AMSTAR score of the meta-analysis (calculated only if this information is indicated in the dataset)

See Also

metaumbrella-package() for the formatting of well-formatted datasets
umbrella() for conducting calculations needed for an umbrella review
add.evidence() for stratifying evidence in an umbrella review

Examples

```
### generate a summary of the results of an umbrella object
summary(umbrella(df.SMD))
```

---

umbrella *Conduct the calculations for an umbrella review*

Description

This function performs various calculations needed for an umbrella review.

Usage

```
umbrella(
  x,
  method.var = "REML",
  mult.level = FALSE,
  r = 0.5,
  pre_post_cor = 0.8,
  method.esb = "TESSPSST",
  true_effect = "UWLS",
  tau2 = NULL,
  max_asymmetry = 10,
  seed = NA,
  verbose = TRUE
)
```

Arguments

x	a well-formatted dataset.
method.var	the estimator used to quantify the between-study variance in the random-effects meta-analysis. Default is the Restricted Likelihood Maximum ("REML") estimator. Alternatively, DerSimonian and Laird "DL", Hartung-Knapp-Sidik-Jonkman "hksj" (applies a Hartung-Knapp-Sidik-Jonkman adjustment on the results of a "DL" estimator), maximum-likelihood "ML" or Paule-Mandel "PM" estimators can be used. A fixed-effect meta-analysis can be obtained by indicated the method.var = "FE" argument.
mult.level	a logical variable indicating the presence of multiple effect sizes per study in at least one factor of the umbrella review. Default is FALSE (i.e., each study of all factors include only one effect size). If mult.level = TRUE is specified, the Borenstein's methods are used to generate only one effect size per study. See metaumbrella-package for more information.
r	a correlation coefficient indicating the strength of the association between multiple outcomes (or time-points) within the same study. The r value is applied to all studies with a "outcomes" value in the reverse_es column that have no indication of correlation in the well-formatted dataset. Default is 0.5.
pre_post_cor	The value of the correlation coefficient between baseline and follow-up scores in pre-post studies. If your umbrella review includes pre-post controlled studies, you should indicate the mean pre-post correlation across groups. Only needed when using the SMC measure.
method.esb	the method used to conduct the excess of statistical significance test. It must be "PSST", "TESS" or "TESSPSST" (see details). Default is TESSPSST.
true_effect	the method to estimate the true effect in the test for excess significance. It must be "largest", "UWLS", "pooled" or a numeric value (see details). Default is "UWLS".
tau2	The tau2 value that should be used when using one of the PSST, TESS or TESSPSST methods (only if you want this value to be different from the one automatically estimated during the meta-analytic calculations).
max_asymmetry	The percentage of assymetry tolerated in the 95% CI of the MD, OR and RR effect measures. Default is 10%. Any stronger asymmetry will stop the analysis.
seed	an integer value used as an argument by the set.seed() function. Only used for the "IT.binom" and "IT.chisq" tests for excess significance with ratios (i.e., "OR", "RR", "IRR" or their logarithm) as effect size measures.
verbose	a logical variable indicating whether text outputs and messages should be generated. We recommend turning this option to FALSE only after having carefully read all the generated messages.

Details

This function automatically performs calculations allowing to stratify evidence according to various criteria. For each factor included in a well-formatted dataset, this function automatically:

- performs fixed- or random-effects meta-analyses.

- provides an estimation of the between-study variance and heterogeneity using three indicators (tau^2, Q-statistic and I^2 statistic).
- estimates the 95% prediction interval (if the number of studies is equal or larger to 3).
- provides an identification of the statistical significance of the largest study included in the meta-analysis.
- provides an assessment of publication bias using the Egger's test (if the number of studies is equal or larger to 3).
- provides an assessment of excess significance using various methods.
- performs a jackknife leave-one-out meta-analysis (if the number of studies is equal or larger to 2).
- calculates the proportion of participants included in studies at low risk of bias (if study quality is indicated in the dataset).

A specificity of the `umbrella()` function is that it does not include arguments to specify the name of the columns of the dataset used as input. Instead, the function requires users to prepare a dataset that meets fixed rules. Details on how building this `well-formatted dataset` are given in the `metaumbrella-package` section of this manual. A vignette also provides a step-by-step tutorial. Moreover, examples of `well-formatted datasets` are available as data distributed along with the package (see df.OR, df.OR.multi, df.R, df.SMC,df.SMD, df.RR, df.HR, df.IRR).

When estimating the test for excess significance, the `umbrella()` function must assume a best approximation of the true effect. The `true_effect` argument can be used to select the method that will be applied to estimate the true effect.

- If `"largest"` is entered, the true effect size is assumed to be equal to the effect size of the largest study included in the meta-analysis.
- If `"pooled"` is entered, the true effect size is assumed to be equal to the meta-analytic pooled effect size.
- If `"UWLS"` is entered, the true effect size is assumed to be equal to unrestricted weighted least squares weighted average.
- If a `numeric` value is entered, the true effect size is assumed to be equal to the value entered by the user (note that the value of ratios must be in their natural scale).

Last, this function performs a statistical test to determine whether the observed number of statistically significant studies is higher than expected given the mean statistical power. The `method.esb` argument can be used to select the test. Details on each method can be found in the `esb.test` section.

Value

The `umbrella()` function returns an object of class "umbrella", which is a list containing information required for stratifying the evidence. This list contains, for each factor included in the umbrella review:

`factor`	the name of the factor analyzed.
`measure`	the measure of the effect used to perform the calculations.

x	the data used to conduct the meta-analysis. Note that these data may be slightly different from the raw data introduced.
x_multi	the original data when there is a multivariate structure. Note that these data may be slightly different from the raw data introduced.
x_shared	a dataframe allowing to visualize adjustments made when a shared_nexp or shared_controls correction is requested (see metaumbrella-package for more information).
n	the overall number of studies, cases and controls.
method.var	the estimator used for fitting the random effects meta-analyses
ma_results	pooled effect size, p-value and 95% confidence interval and prediction interval of the meta-analysis.
largest	95% confidence interval of the largest study.
heterogeneity	tau^2, I^2 and results of the Q-test.
egger	estimate and p-value of the Egger's test for publication bias.
esb	results of the test for excess significance bias. See esb.test() for more information.
overall_rob	for the overall RoB, the proportion of participants in studies at low risk of bias (weighted mean based on m
report_rob	for the selective reporting bias, the proportion of participants in studies at low risk of bias (weighted mean l
rob	for each individual RoB, the proportion of participants in studies at low risk of bias (weighted mean based
weights	the weights assigned to each study in the meta-analysis.
amstar	AMSTAR score obtained by the meta-analysis.
evidence	evidence class according to some criteria.

The functions print and summary may be used to print the details or a summary of the results.

References

Fusar-Poli, P., Radua, J. (2018). Ten simple rules for conducting umbrella reviews. *Evidence-Based Mental Health*, **21**, 95–100.

Radua, J., Ramella-Cravaro, V., Ioannidis, J.P.A., Reichenberg, A., Phiphopthatsanee, N., Amir, T., Yenn Thoo, H., Oliver, D., Davies, C., Morgan, C., McGuire, P., Murray, R.M., Fusar-Poli, P. (2018) What causes psychosis? An umbrella review of risk and protective factors. *World Psychiatry*, **17**, 49–66.

See Also

metaumbrella-package for the formatting of well-formatted datasets
add.evidence() for stratifying the evidence in an umbrella review
forest() for drawing a forest plot of the factors included in an umbrella review
subset.umbrella() for retrieving a subset of the factors included in an umbrella review
union.umbrella() for combining the factors included in two umbrella reviews

Examples

```
### Perform an umbrella review with random-effects meta-analyses
### with a Hartung-Knapp-Sidik-Jonkman estimator
umb <- umbrella(df.IRR, method.var = "hksj")

### obtain the results of the calculations in a dataframe
summary(umb)

### manually inspect the results of the umbrella review calculations for the 'Smoking' factor
### included in the dataset.
umb$Smoking

### Perform a meta-analysis with multilevel data, assuming a correlation of 0.8
### between all outcomes of the same study
umb.multi <- umbrella(df.OR.multi, mult.level = TRUE, r = 0.8)

### obtain a stratification of the evidence according to the Ioannidis classification
add.evidence(umb.multi, criteria = "Ioannidis")
```

---

umbrella.gui *Graphical User Interface for conducting an umbrella review*

Description

The umbrella.gui function allows to perform an umbrella review in R with a graphical user interface and to generate results outside of R.

Usage

```
umbrella.gui()
```

Details

This function implicitly calls the view.errors.umbrella() to check the formatting of the data, the umbrella() function to perform calculations, the add.evidence() function to stratify evidence (limited to "Ioannidis" and "GRADE" classifications for now) and the forest() function to generate a visual representation of the results.

To use this function:

- The first step is to type umbrella.gui() in the console. This will open a pop-up window.

- The second step is to load a well-formatted dataset (stored in an excel file) by clicking on the button at the right of the "Excel file" label. An exploratory window will open, allowing to select the file that should be uploaded.
- The third step is to select the folder where the results will be exported. Again, this can be achieved by clicking on the button at the right of the "Output folder" label, and selecting the appropriate folder thanks to the exploratory window.
- Last, the classification that should be used for stratifying the evidence has to be selected. For now, "Ioannidis" and "GRADE" classifications are available.

Value

The `umbrella.gui()` function returns several elements including

`csv`	a csv file containing the results of the umbrella review.
`pdf`	a pdf file containing the plot of the results (only if requested, i.e., the `plot_results` argument is set as TRUE).
`html`	an HTML file containing the results of the umbrella review.

See Also

`umbrella()`
`add.evidence()`
`summary.umbrella()`

Examples

```
if(interactive()){
### open the GUI
umbrella.gui()
}
```

`umbrella.xls` *Automatically conduct an umbrella review and export results outside of R*

Description

The `umbrella.xls` function allows to automatically perform an umbrella review and to export results outside of R.

Usage

```
umbrella.xls(
  input_file = NULL,
  sheet = 1,
  output_path = "",
  output_name = "",
  factors_to_analyze = "",
  evidence = NULL,
  plot_results = FALSE
)
```

Arguments

input_file	path indicating the folder where the dataset (stored in an excel file) is located.
sheet	the number of the excel sheet where the data are located.
output_path	path indicating the folder where the results to be generated.
output_name	name of the files created by the function.
factors_to_analyze	the name(s) of factors to analyze. Can be either a character string indicating the name of the factor to analyze or a vector indicating the names of the factors to analyze. By default, all the factors are analyzed.
evidence	criteria used to stratify evidence. Must be either "Ioannidis" or "GRADE".
plot_results	indicate whether a forest plot of the results is returned. Must be either TRUE or FALSE.

Details

This function implicitly calls the `view.errors.umbrella()` to check the formatting of the data, the `umbrella()` function to perform calculations, the `add.evidence()` function to stratify evidence (limited to "Ioannidis" and "GRADE" classifications for now) and the `forest()` function to generate a visual representation of the results.

Value

The `umbrella.xls()` function returns several elements including

csv	a csv file containing the results of the umbrella review.
pdf	a pdf file containing the plot of the results (only if requested, i.e., the `plot_results` argument is set as TRUE).
html	an HTML file containing the results of the umbrella review.

See Also

`umbrella()`
`add.evidence()`
`summary.umbrella()`

Examples

```
if(interactive()){
### perform an umbrella review according to the GRADE criteria.
### the umbrella review is restricted to the factor "Pharmacological"
### Note that the df.SMD should be stored under a .xls or .xlsx format
umbrella.xls(input_file = file.choose(),
             sheet = 1,
             output_path = choose.dir(),
             output_name = "Pharmacological_Ioannidis",
             factors_to_analyze = c("Pharmacological"),
             evidence = "GRADE",
             plot_results = TRUE)
}
```

---

union.umbrella *Union of two objects of class "umbrella"*

Description

Combine the factors included in two umbrella objects

Usage

```
union.umbrella(x, y, ...)
```

Arguments

x	an object of class "umbrella".
y	an object of class "umbrella".
...	other arguments that can be passed to the function

Details

This function allows to combine the results of two objects of class "umbrella". This function is particularly useful when different specifications are used to analyze different factors. It is not possible to union two objects of class "umbrella" with different classifications.

Value

Return an object of class"umbrella", with the factors of the two objects of class "umbrella".

Examples

```
### union raw umbrella objects
umb1 <- umbrella(df.SMD, method.var = "REML")
umb2 <- umbrella(df.OR, method.var = "PM")
umb.union <- union.umbrella(umb1, umb2)
summary(add.evidence(umb.union, criteria = "GRADE"))

### union umbrella objects after applying stratification of evidence
umb1 <- add.evidence(umbrella(df.SMD), criteria = "GRADE")
umb2 <- add.evidence(umbrella(df.OR), criteria = "GRADE")
umb3 <- add.evidence(umbrella(df.IRR), criteria = "GRADE")
umb.union <- union.umbrella(union.umbrella(umb1, umb2), umb3)
summary(umb.union)
```

---

view.errors.umbrella *Detect incorrect formatting of a dataset*

Description

Check the formatting of a dataset to ensure it can be passed to the functions of the **metaumbrella** package.

Usage

```
view.errors.umbrella(data, return = "data_and_messages")
```

Arguments

`data`	a dataframe
`return`	the type of information returned by the function. Must be either "messages", "data_and_messages", or "data".

Details

The functions included in the **metaumbrella** package require very specific formatting of the dataset (see `metaumbrella-package`). The `view.errors.umbrella()` function checks that a dataframe meets all requirements of the functions of the **metaumbrella** package. If this function finds some formatting issues, error messages describing the issues are produced and the rows / columns in which the issues occurred are identified.

Value

Depending on the value passed to the `return` argument, different information is returned:

`"messages"`	return global messages describing the different formatting issues.
`"data"`	return the rows of the original dataset with formatting issues (see below).

`"data_and_messages"` return both (i) global messages describing the different formatting issues and (ii) the rows of the original dataset with formatting issues (see below).

When returning a dataset (i.e., when `"data"` or `"data_and_messages"` are indicated in the return argument), the rows with problematic formatting are identified and two new columns are added to the original dataset (`column_type_errors` and `column_errors`). These columns help to understand formatting issues.

- A **WARNING** value in the `column_type_errors` column indicates a potential issue that should be checked but that do not prevent calculations.
- An **ERROR** value in the `column_type_errors` column indicates an issue that must be solved before running calculations.
- The text in the `column_errors` describes the issues encountered for each problematic row.

Examples

```
df.errors1 <- df.errors2 <- df.errors3 <- df.errors4 <- df.OR

### include some unknown measures
df.errors1$measure[c(1,4,12)] <- "unknown_measure"
view.errors.umbrella(df.errors1, return = "data_and_messages")

### include some not numeric inputs while expected
df.errors2$value[c(2,13,15)] <- c("a", "b", "c")
view.errors.umbrella(df.errors2, return = "data")

### make the lower bound of a confidence interval > to the value
df.errors3$ci_lo[c(12,14,21)] <- c(5,6,7)
view.errors.umbrella(df.errors3, return = "messages")

### create errors in sample sizes
df.errors4$n_cases_exp[c(5,10,15)] <- c(100, 200, 300)
view.errors.umbrella(df.errors4, return = "data_and_messages")
```

Index