Package ‘metaumbrella’

October 13, 2022

Type Package
Title Umbrella Review Package for R
Version 1.0.5
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, 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 7.1.2
VignetteBuilder knitr
Language en-US
NeedsCompilation no
Repository CRAN
Date/Publication 2022-06-13 06:50:09 UTC
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-formed dataset). We present here the rules that must be respected when creating a well-formed 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-formed 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).

1. "**SMD**":

<table>
<thead>
<tr>
<th>mean_cases + mean_controls +</th>
<th>n_cases + n_controls</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo + ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>sd_cases + sd_controls</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

2. "**G**":

<table>
<thead>
<tr>
<th>n_cases + n_controls</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo + ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

3. "**MD**":

<table>
<thead>
<tr>
<th>n_cases + n_controls</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo + ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>-</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

4. "**SMC**":

<table>
<thead>
<tr>
<th>mean_pre_cases +</th>
<th>mean_pre_controls +</th>
</tr>
</thead>
<tbody>
<tr>
<td>sd_pre_cases +</td>
<td>sd_pre_controls +</td>
</tr>
<tr>
<td>mean_cases +</td>
<td>mean_controls +</td>
</tr>
<tr>
<td>sd_cases +</td>
<td>sd_controls +</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>pre_post_cor</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

5. "R":

<table>
<thead>
<tr>
<th>n_sample</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo</th>
<th>ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_sample</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
<tr>
<td>n_sample</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
</tbody>
</table>

6. "Z":

<table>
<thead>
<tr>
<th>n_sample</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo</th>
<th>ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_sample</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
<tr>
<td>n_sample</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
</tbody>
</table>

7. "OR" or "logOR":

<table>
<thead>
<tr>
<th>n_cases_exp + n_controls_exp + n_cases_nexp</th>
<th>n_exp + n_nexp</th>
<th>n_cases + n_controls</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo</th>
<th>ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_cases_exp + n_controls_exp + n_cases_nexp</td>
<td>n_exp + n_nexp</td>
<td>n_cases + n_controls</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
<tr>
<td>n_cases_exp + n_controls_exp + n_cases_nexp</td>
<td>n_exp + n_nexp</td>
<td>n_cases + n_controls</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
</tbody>
</table>

8. "RR" or "logRR":

<table>
<thead>
<tr>
<th>n_cases_exp + n_controls_exp + n_cases_nexp</th>
<th>n_exp + n_nexp</th>
<th>n_cases + n_controls</th>
<th>value</th>
<th>se</th>
<th>var</th>
<th>ci_lo</th>
<th>ci_up</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_cases_exp + n_controls_exp + n_cases_nexp</td>
<td>n_exp + n_nexp</td>
<td>n_cases + n_controls</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
<tr>
<td>n_cases_exp + n_controls_exp + n_cases_nexp</td>
<td>n_exp + n_nexp</td>
<td>n_cases + n_controls</td>
<td>value</td>
<td>se</td>
<td>var</td>
<td>ci_lo</td>
<td>ci_up</td>
</tr>
</tbody>
</table>
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 \texttt{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 \texttt{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.

- **shared_nexp**: In some situations, several studies share participants from the same non-exposed group but compare this group to various exposed groups. When several studies in the same factor share a same non-exposed group, they should be identified as such by having the same shared_nexp value. Identifying studies sharing the same non-exposed group allows to adjust calculations (the size of the shared sample is divided by the number of studies sharing the sample). Studies not sharing their non-exposed group should have a NA (or a unique) value in the shared_nexp column.

- **shared_controls**: In some situations, several studies share participants from the same control group but compare this group to various experimental groups. When several studies in the same factor share a same control group, they should be identified as such by having the same shared_controls value. Identifying studies sharing the same control group allows to adjust calculations (the size of the shared sample is divided by the number of studies sharing the sample). Studies not sharing their control group should have a NA (or a unique) value in the shared_controls column.

- **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 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.

- **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.
add.evidence

Usage

add.evidence(
  x,
  criteria = "Ioannidis",
  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),
  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),
  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".

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.

"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 following criteria:

- Imprecision: a total number of participants included in the meta-analysis giving a lower power than 0.8 to detect a SMD = 0.20 leads to a downgrading of 1 class. A number of participants giving a lower power than 0.8 to detect a SMD = 0.50, leads to a downgrading of 2 classes.

- Limitations: a proportion of participants included in studies at low risk of bias inferior to 75% leads to a downgrading of 1 class. A proportion inferior to 50% leads to a downgrading of 2 classes.

- Publication bias: a p-value of an Egger’s test < .10 leads to a downgrading of 1 class.

- Inconsistency: an $I^2$ value >= 0.5 leads to a downgrading of 1 class.

"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**


**See Also**

umbrella() for conducting an umbrella review.

**Examples**

```r
### 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).
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.
- n_controls: number of controls.

Source
No source, the data are entirely fictitious.
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_cases</td>
<td>number of cases (sum of the number of cases in the exposed and non-exposed groups).</td>
</tr>
<tr>
<td>n_cases_exp</td>
<td>number of cases in the exposed group.</td>
</tr>
<tr>
<td>n_cases_nexp</td>
<td>number of cases in the non-exposed group.</td>
</tr>
<tr>
<td>time</td>
<td>total person-time at risk (sum of the person-time at risk in the exposed and non-exposed groups).</td>
</tr>
<tr>
<td>time_exp</td>
<td>person-time at risk in the exposed group.</td>
</tr>
<tr>
<td>time_nexp</td>
<td>person-time at risk in the non-exposed group.</td>
</tr>
</tbody>
</table>

**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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>meta_review</td>
<td>name of the first author of the meta-analysis.</td>
</tr>
<tr>
<td>factor</td>
<td>name of the neurodevelopmental disorders on which the effect of the risk factor is studied</td>
</tr>
<tr>
<td>author</td>
<td>first study author of the individual studies.</td>
</tr>
<tr>
<td>year</td>
<td>year of publication of the individual studies.</td>
</tr>
<tr>
<td>measure</td>
<td>type of effect size (OR).</td>
</tr>
</tbody>
</table>
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.
mean_controls  means of participants in the control arm.
sd_cases  standard deviation of participants in the experimental arm.
sd_controls  standard deviation of participants in the control arm.
amstar  AMSTAR score of the meta-analysis

Source
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
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.

**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.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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>meta_review</td>
<td>name of the first author of the meta-analysis.</td>
</tr>
<tr>
<td>factor</td>
<td>name of the intervention studied.</td>
</tr>
<tr>
<td>author</td>
<td>first study author of the individual studies.</td>
</tr>
<tr>
<td>year</td>
<td>year of publication of the individual studies.</td>
</tr>
<tr>
<td>measure</td>
<td>type of effect size (SMD).</td>
</tr>
</tbody>
</table>
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.
### df.train

<table>
<thead>
<tr>
<th><strong>Name</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>risk_factor</td>
<td>name of the intervention studied.</td>
</tr>
<tr>
<td>author_study</td>
<td>first study author of the individual studies.</td>
</tr>
<tr>
<td>year_publication_study</td>
<td>year of publication of the individual studies.</td>
</tr>
<tr>
<td>type_of_effect_size</td>
<td>type of effect size.</td>
</tr>
<tr>
<td>number_of_cases_exposed</td>
<td>number of cases in the exposed group.</td>
</tr>
<tr>
<td>number_of_cases_non_exposed</td>
<td>number of cases in the non-exposed group.</td>
</tr>
<tr>
<td>number_of_controls_exposed</td>
<td>number of controls in the exposed group.</td>
</tr>
<tr>
<td>number_of_controls_non_exposed</td>
<td>number of controls in the non-exposed group.</td>
</tr>
<tr>
<td>number_of_participants_exposed</td>
<td>total number of participants in the exposed group.</td>
</tr>
<tr>
<td>number_of_participants_non_exposed</td>
<td>total number of participants in the non-exposed group.</td>
</tr>
<tr>
<td>number_of_cases</td>
<td>number of cases.</td>
</tr>
<tr>
<td>number_of_controls</td>
<td>number of controls.</td>
</tr>
<tr>
<td>effect_size_value</td>
<td>value of the effect size</td>
</tr>
<tr>
<td>low_bound_ci</td>
<td>lower bound of the 95% confidence interval.</td>
</tr>
<tr>
<td>up_bound_ci</td>
<td>upper bound of the 95% confidence interval.</td>
</tr>
<tr>
<td>time_disease_free</td>
<td>total person-time at risk (sum of the person-time at risk in the exposed and non-exposed groups).</td>
</tr>
<tr>
<td>mean_of_intervention_group</td>
<td>mean of the intervention group</td>
</tr>
<tr>
<td>mean_of_control_group</td>
<td>mean of the control group</td>
</tr>
<tr>
<td>sd_of_intervention_group</td>
<td>sd of the intervention group</td>
</tr>
<tr>
<td>sd_of_control_group</td>
<td>sd of the control group</td>
</tr>
</tbody>
</table>

### 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


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 = NA
)

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 IT.binom, IT.chisq, 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 "IT.binom" is entered, the excess statistical significance test described by Ioannidis and Trikalinos (2007) is performed using a binomial exact test. This test explores whether the number of studies with statistically significant results is higher than what could have been expected given the mean statistical power to detect the best approximation of the true effect.
* If "IT.chisq" is entered, the excess statistical significance test described by Ioannidis and Trikalinos (2007) is performed using a chi-square test. This test explores whether the number of studies with statistically significant results is higher than what could have been expected given the mean statistical power to detect the best approximation of the true effect.
* 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


Examples

```r
### 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 = "IT.binom", true_effect = "largest")

### perform an excess significance bias using the umbrella function
esb.umbrella <- umbrella(df, method.esb = "IT.binom", 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$n_controls)
metafor::rma(yi = G$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 = "IT.binom")
```
### perform an excess significance bias on a meta object

```r
meta <- meta::metagen(TE = G$value, seTE = G$se,
                      sm = "SMD",
                      n.e = df$n_cases,
                      n.c = df$n_controls,
                      data = df)
```

```r
esb.meta <- esb.test(meta, input = "meta", method.esb = "IT.binom")
```

```r
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”*

Description

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

Usage

```r
forest(x, ...)
```

Arguments

- `x` an object of class “umbrella”
- `...` additional arguments that can be passed to this function

Details

For now, this function simply applies the `forest.umbrella()` function on an object of class “umbrella”.

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

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

Draw a forest plot of the factors included in an umbrella review. The basic version of this plot contains three parts:

- a column displaying the name of the factors (referred to as the ‘factor’ column).
- a graphical representation of the pooled effect sizes and 95% CI of the factors.
- a column displaying the numeric values of the pooled effect size and 95% CI of the factors (referred to as the ‘value’ column).

Usage

```r
## S3 method for class 'umbrella'
forest(
  x,
  measure = "eG",
  main_title = NA,
  main_value = NA,
  main_x_axis = NA,
  max.value = NULL,
  pch = "square",
  print.classes = NULL,
  col_sig = c(NA, NA),
  log_cex_dots = FALSE,
  fix_size_dots = NA,
  xlim = NULL,
  xlim_main_title = 0,
  xlim_value = 2.1,
  xlim_factor = -2.1,
  ylim_correction_value = 0,
  ylim_correction_text = 0,
  cex_title = 1.4,
  cex_text_header = 1,
  cex_text = 0.9,
  cex_value_header = 1,
  cex_value = 0.9,
  cex_x_axis = 1.1,
  cex_x_axis_value = 0.8,
  cex_dots = 1,
  col_title = "#1D1D1D",
  col_text_header = "#252525",
  col_text = "#252525",
  col_value_header = "#252525",
)```
col_value = "#252525",
col_x_axis = "#252525",
col_dots = "#252525",
col_border = "#000000",
col_lines = "#252525",
pos_value = "left-align",
pos_text = "right-align",
add_columns = NULL,
main_add_columns = NA,
xlim_add_columns = NA,
x_lim_adj = 0,
y_lim_adj = 0,
x_axis_adj = 0,
)

Arguments

x an object of class “umbrella”.

measure type of effect size used in the plot. Default is equivalent Hedges’ g ("eG"). Alternatively, equivalent Odds Ratio ("eOR") can be used.

main_title the title of the plot

main_value a title for the header of the 'value' column.

main_x_axis title of the x-axis

max.value the maximum value that will be plotted on the x-axis. Must be strictly superior to 1 when equivalent odds ratio (eOR) measure is used, and strictly superior to 0 when SMD measure is used. Note that minimum value will be automatically set to the inverse of max.value for eOR measure and to - max.value for SMD measure.

pch the shape used to depict the pooled effect size (must be either "square", "circle", "rhombus" or "triangle")

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
- "High", "Moderate", "Weak" and/or "Very weak" for the "GRADE" classification
- "I", "II", "III", "IV", and/or "V" for the "Personalized" classification

col_sig a vector of two colors. Statistically significant factors are displayed using these colors:
- Protective factors (i.e., with an eOR < 1 or an SMD < 0) are displayed with the first color
- Risk factors (i.e., with an eOR >= 1 or an SMD >= 0) will be displayed with the second color.
- Non-significant factors are displayed with the color indicated in the ‘col_dots’ argument.
log_cex_dots  a logical value indicating whether the logarithm of the size of the dot should be
used (should be TRUE or FALSE). When set as TRUE, this argument allows to
reduce the disparities in the size of the dots between factors.

fix_size_dots  a unique positive numeric value that is used to fix the size of all dots

xlim  a vector containing the limits of the x-axis (x1, x2) of the plot.

xlim_main_title  position of the title of the plot on the x-axis.

xlim_value  position of the 'value' column on the x-axis.

xlim_factor  position of the 'factor' column on the x-axis.

ylim_correction_value  when the position of columns is modified (e.g., "right-align" versus "center"),
the height of the columns can be slightly modified. This correction value allows
to adjust the height of the 'value' column.

ylim_correction_text  when the position of columns is modified (e.g., "right-align" versus "center"),
the height of the columns can be slightly modified. This correction value allows
to adjust the height of the 'factor' column and to any new column added to the
plot.

cex_title  numeric value indicating the amount by which title text should be scaled relative
to the default.

cex_text_header  numeric value indicating the amount by which the text of the header of the 'factor'
column and of any column added to the plot should be scaled relative to the
default.

cex_text  numeric value indicating the amount by which the text of the 'factor' column and of any column added to the plot should be scaled relative to the default.

cex_value_header  numeric value indicating the amount by which the text of the header of the 'value'
column should be scaled relative to the default.

cex_value  numeric value indicating the amount by which the text of the 'value' column
should be scaled relative to the default.

cex_x_axis  numeric value indicating the amount by which the text of the x-axis title should
be scaled relative to the default.

cex_x_axis_value  numeric value indicating the amount by which the text of the x-axis values
should be scaled relative to the default.

cex_dots  numeric value indicating the amount by which dot size should be scaled relative
to the default.

col_title  color of the font of the title.

col_text_header  color of the font of the header of the 'factor' column and to any column added
to the plot.

col_text  color of the font of the 'factor' column and of any column added to the plot.
col_value_header  
color of the header of the 'value' column.

col_value      
color of the font of the 'value' column.

col_x_axis     
color of the font of the x-axis.

col_dots       
color of the dots.

col_border     
color of the border of the dots.

col_lines      
color of the lines.

pos_value      
type of centering of the column displaying the values. Must be either "left-align", "center", or "right-align".

pos_text       
type of centering of the 'factor' column. Must be either "left-align", "center", or "right-align".

add_columns    
a vector/dataframe containing the columns that should be added to the plot. The number of rows of the columns added to the plot should be equal to the number of factors.

main_add_columns  
a vector containing the name of the header of the new columns added to the plot (default is the name of the columns in the dataset).

xlim_add_columns 
position of the new columns added on x-axis. Should be a numeric vector containing the exact same number values as the number of columns added. Mandatory when columns are added to the plot.

x_lim_adj      
a numeric value that can be used to adjust the size of the window of the plot on the x-axis.

y_lim_adj      
a numeric value that can be used to adjust the size of the window of the plot on the y-axis.

x_axis_adj      
a numeric value that can be used to adjust the height of the space between the x-axis and the title of the x-axis.

...            
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

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().
print.umbrella

References


Examples

```r
### perform an umbrella review of various datasets
umbs <- union.umbrella(union.umbrella(union.umbrella(union.umbrella(
  umbrella(df.OR), umbrella(df.SMD)), umbrella(df.RR)), umbrella(df.IRR)), umbrella(df.HR))

### stratify the evidence according to some Personalized criteria
umbs.crit <- add.evidence(umbs, criteria = "Personalized",
  class_I = c(n_cases = 1000, p_value = 1e-6, esb_p = 0.10, egger_p = 0.10,
    I2 = 0.25, pi = "notnull"),
  class_II = c(n_cases = 1000, p_value = 1e-6, largest_CI = "notnull"),
  class_III = c(n_cases = 1000, p_value = 1e-3),
  class_IV = c(p_value = 5e-2))
sum.umbs <- summary.umbrella(umbs.crit)

### generate a forest plot of each factor included in the umbrella review
forest.umbrella(umbs.crit, max.value = 7,
  main_title = "Plot of the umbrella review")
```

print.umbrella

*Print a summary of an object of class “umbrella”*

Description

Print a summary of an object of class “umbrella”

Usage

```r
## 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

```r
### 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

```r
# 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

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

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

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**

```r
## S3 method for class 'umbrella'
summary(object, digits = 3, 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.
- `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:

<table>
<thead>
<tr>
<th>Factor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>the class assigned during the stratification of evidence.</td>
</tr>
<tr>
<td>n_studies</td>
<td>the number of independent studies included in the factor.</td>
</tr>
<tr>
<td>total_n</td>
<td>the total number of participants included in the factor.</td>
</tr>
<tr>
<td>n_cases</td>
<td>the number of cases included in the factor.</td>
</tr>
<tr>
<td>n_controls</td>
<td>the number of controls included in the factor.</td>
</tr>
<tr>
<td>measure</td>
<td>the measured used in the calculations.</td>
</tr>
<tr>
<td>value</td>
<td>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.</td>
</tr>
<tr>
<td><strong>value_CI</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>eG</strong></td>
<td>the value of the pooled effect size expressed in equivalent Hedges’ g (eG).</td>
</tr>
<tr>
<td><strong>eG_CI</strong></td>
<td>the 95% CI around the pooled effect size expressed in eG.</td>
</tr>
<tr>
<td><strong>eOR</strong></td>
<td>the value of the pooled effect size expressed in equivalent Odds ratio (eOR).</td>
</tr>
<tr>
<td><strong>eOR_CI</strong></td>
<td>the 95% CI around the pooled effect size expressed in eOR.</td>
</tr>
<tr>
<td><strong>p_value</strong></td>
<td>the p-value of the pooled effect size.</td>
</tr>
<tr>
<td><strong>I2</strong></td>
<td>the inconsistency ($I^2$) value (calculated only if the number of studies in the meta-analysis is equal or larger to 2).</td>
</tr>
<tr>
<td><strong>PI_eG</strong></td>
<td>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).</td>
</tr>
<tr>
<td><strong>PI_eOR</strong></td>
<td>the 95% PI expressed in eOR (calculated only if the number of studies in the meta-analysis is equal or larger to 3).</td>
</tr>
<tr>
<td><strong>PI_sign</strong></td>
<td>whether the 95% PI includes the null value (&quot;nonnull&quot; vs. &quot;null&quot;).</td>
</tr>
<tr>
<td><strong>egger_p</strong></td>
<td>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).</td>
</tr>
<tr>
<td><strong>egger_sign</strong></td>
<td>whether the p-value of the Egger’s test is &lt; .05 (&quot;sig.&quot; vs. &quot;ns&quot;).</td>
</tr>
<tr>
<td><strong>ESB_p</strong></td>
<td>the p-value of the test for excess of significance bias.</td>
</tr>
<tr>
<td><strong>ESB_sign</strong></td>
<td>whether the p-value of the excess of significance test is &lt; .05 (&quot;sig.&quot; vs. &quot;ns&quot;).</td>
</tr>
<tr>
<td><strong>power_med</strong></td>
<td>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).</td>
</tr>
<tr>
<td><strong>power</strong></td>
<td>present only in the 'Personalized' classification.</td>
</tr>
<tr>
<td>- 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.</td>
<td></td>
</tr>
<tr>
<td>- 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.</td>
<td></td>
</tr>
</tbody>
</table>
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,
)
method.esb = "TESSPSST",
true_effect = "UWLS",
pre_post_cor = NA,
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 metaunderella-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.

method.esb the method used to conduct the excess of statistical significance test. It must be "IT.binom", "IT.chisq", "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".

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.

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.
the data used to conduct the meta-analysis. Note that these data may be slightly different from the raw data introduced.

the original data when there is a multivariate structure. Note that these data may be slightly different from the raw data introduced.

dataframe allowing to visualize adjustments made when a shared_nexp or shared_controls correction is requested (see metaumbrella-package for more information).

the overall number of studies, cases and controls.

the estimator used for fitting the random effects meta-analyses

pooled effect size, p-value and 95% confidence interval and prediction interval of the meta-analysis.

95% confidence interval of the largest study.

tau, I² and results of the Q-test.

estimate and p-value of the Egger’s test for publication bias.

results of the test for excess significance bias. See esb.test() for more information.

percentage of participants in studies at low risk of bias.

AMSTAR score obtained by the meta-analysis.

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


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

```r
### 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")
```

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()
}

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()
union.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”.

add.evidence()
summary.umbrella()
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

```r
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

* datasets
  df.HR, 12
  df.IRR, 13
  df.OR, 14
  df.OR.multi, 15
  df.R, 16
  df.radua2019, 17
  df.RR, 19
  df.SMC, 20
  df.SMD, 21
  df.train, 22

  add.evidence, 2, 3, 8, 37, 40, 42, 44
  df.HR, 3, 12, 39
  df.IRR, 3, 13, 39
  df.OR, 3, 14, 39
  df.OR.multi, 7, 15, 39
  df.R, 3, 16, 39
  df.radua2019, 17
  df.RR, 3, 19, 39
  df.SMC, 3, 20, 39
  df.SMD, 3, 21, 39
  df.train, 22
  drop.evidence, 24

  esb.test, 25, 39, 40

  forest, 2, 3, 28, 41
  forest.umbrella, 28, 29

  metaumbrella-package, 2, 40

  print.umbrella, 33

  subset.umbrella, 34, 41
  summary.umbrella, 34, 35, 42, 44

  umbrella, 2, 3, 7, 11, 24, 37, 39, 42, 43
  umbrella.gui, 41
  umbrella.xls, 42

union.umbrella, 41, 44
view.errors.umbrella, 45