# Package 'metafor'

January 27, 2015

**Version** 1.9-5

**Date** 2014-11-24

**Title** Meta-Analysis Package for R

**Depends** R (>= 3.0.0)

**Imports** stats, utils, Matrix

**Suggests** R.rsp, testthat, Formula, lme4, numDeriv, polycor, BiasedUrn,
survival, Epi, minqa, multcomp, plyr, igraph, CompQuadForm,
nloptr

**Description** The metafor package provides a comprehensive collection of functions for conduct-
ing meta-analyses in R. The package includes functions to calculate various effect sizes or out-
come measures, fit fixed-, random-, and mixed-effects models to such data, carry out modera-
tor and meta-regression analyses, and create various types of meta-analytical plots (e.g., for-
est, funnel, radial, L'Abbe, Baujat plots). For meta-analyses of binomial and person-
time data, the package also provides functions that implement specialized methods, includ-
ing the Mantel-Haenszel method, Peto's method, and a variety of suitable generalized lin-
ear (mixed-effects) models (i.e., mixed-effects (conditional) logistic and Poisson regression mod-
els). Finally, the package provides functionality for fitting meta-analytic multivari-
ate/multilevel models that account for non-independent sampling errors and/or true ef-
fects (e.g., due to the inclusion of multiple treatment studies, multiple end-
points, or other forms of clustering). Network meta-analyses and meta-analyses account-
ing for known correlation structures (e.g., due to phylogenetic relatedness) can also be conducted.

**License** GPL (>= 2)

**ByteCompile** TRUE

**LazyData** TRUE

**Encoding** UTF-8

**VignetteBuilder** R.rsp

**URL** http://www.metafor-project.org http://www.wvbauer.com

**Author** Wolfgang Viechtbauer [aut, cre]

**Maintainer** Wolfgang Viechtbauer <wvb@metafor-project.org>

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-11-24 16:07:16

# R **topics documented:**

---

metafor-package · *Metafor: A Meta-Analysis Package for R*

---

### Description

The **metafor** package provides a comprehensive collection of functions for conducting meta-analyses in R. The package includes functions for calculating various effect size or outcome measures frequently used in meta-analyses (e.g., risk differences, risk ratios, odds ratios, standardized mean differences, Fisher's r-to-z-transformed correlation coefficients) and then allows the user to fit fixed-, random-, and mixed-effects models to these data. By including study-level covariates ('moderators') in these models, so-called 'meta-regression' analyses can be carried out. For meta-analyses of $2 \times 2$ tables, proportions, incidence rates, and incidence rate ratios, the package also provides functions that implement specialized methods, including the Mantel-Haenszel method, Peto's method, and a variety of suitable generalized linear (mixed-effects) models (i.e., mixed-effects (conditional) logistic and Poisson regression models). For non-independent effect sizes or outcomes (e.g., due to correlated sampling errors, correlated true effects or outcomes, or other forms of clustering), the package also provides a function for fitting multilevel/multivariate meta-analytic models.

Various methods are available to assess model fit, to identify outliers and/or influential studies, and for conducting sensitivity analyses (e.g., standardized residuals, Cook's distances, leave-one-out analyses). Advanced techniques for hypothesis tests and obtaining confidence intervals (e.g., for the average effect size or for the model coefficients in a meta-regression model) have also been implemented (e.g., the Knapp and Hartung method, permutation tests).

The package also provides functions for creating forest, funnel, radial (Galbraith), normal quantile-quantile, L'Abbé, and Baujat plots. The presence of funnel plot asymmetry (which may be indicative of publication bias) and its potential impact on the results can be examined via the rank correlation and Egger's regression test and by applying the trim and fill method.

### The rma.uni Function

The various meta-analytic models that are typically used in practice are special cases of the general linear (mixed-effects) model. The `rma.uni` function (with alias `rma`) provides a general framework for fitting such models. The function can be used in conjunction with any of the usual effect size or outcome measures used in meta-analyses (e.g., log odds ratios, log relative risks, risk differences, mean differences, standardized mean differences, raw correlation coefficients, correlation coefficients transformed with Fisher's r-to-z transformation, and so on). For details on these effect size or outcome measures, see the documentation of the `escalc` function. The notation and models underlying the `rma.uni` function are explained below.

For a set of $i = 1, \ldots, k$ independent studies, let $y_i$ denote the observed value of the effect size or outcome measure in the $i$th study. Let $\theta_i$ denote the corresponding (unknown) true effect or outcome, such that

$$y_i | \theta_i \sim N(\theta_i, v_i).$$

In other words, the observed effects or outcomes are assumed to be unbiased and normally distributed estimates of the corresponding true effects or outcomes with sampling variances equal to $v_i$. The $v_i$ values are assumed to be known. Depending on the outcome measure used, a bias correction, normalizing, and/or variance stabilizing transformation may be necessary to ensure that these

assumptions are (approximately) true (e.g., the log transformation for odds ratios, the bias correction for standardized mean differences, Fisher's r-to-z transformation for correlations; see escalc for more details).

The **fixed-effects model** conditions on the true effects/outcomes and therefore provides a *conditional inference* about the set of $k$ studies included in the meta-analysis. When using weighted estimation, this implies that the fitted model provides an estimate of

$$\bar{\theta}_w = \sum_{i=1}^{k} w_i \theta_i / \sum_{i=1}^{k} w_i,$$

that is, the *weighted average* of the true effects/outcomes in the set of $k$ studies, with weights equal to $w_i = 1/v_i$ (this is what is often described as the 'inverse-variance' method in the meta-analytic literature). One can also employ an unweighted estimation method, which provides an estimate of the *unweighted average* of the true effects/outcomes in the set of $k$ studies, that is, an estimate of

$$\bar{\theta}_u = \sum_{i=1}^{k} \theta_i / k.$$

For weighted estimation, one could also choose to estimate $\bar{\theta}_w$, where the $w_i$ values are user-defined weights (inverse-variance weights or unity weights as in unweighted estimation are just special cases). It is up to the user to decide to what extent $\bar{\theta}_w$ is a meaningful parameter to estimate (regardless of the weights used).

Moderators can be included in the fixed-effects model, yielding a **fixed-effects with moderators model**. Again, since the model conditions on the set of $k$ studies included in the meta-analysis, the regression coefficients from the fitted model estimate the weighted relationship (in the least squares sense) between the true effects/outcomes and the moderator variables within the set of $k$ studies included in the meta-analysis (again using weights equal to $w_i = 1/v_i$). The unweighted relationship between the true effects/outcomes and the moderator variables can be estimated when using the unweighted estimation method. Again, user-defined weights could also be used.

The **random-effects model** does not condition on the true effects/outcomes. Instead, the $k$ studies included in the meta-analysis are assumed to be a random selection from a larger population of studies. In rare cases, the studies included in a meta-analysis are actually sampled from a larger collection of studies. More typically, the population of studies is a hypothetical population of an essentially infinite set of studies comprising all of the studies that have been conducted, that could have been conducted, or that may be conducted in the future. We assume that $\theta_i \sim N(\mu, \tau^2)$, that is, the true effects/outcomes in the population of studies are normally distributed with $\mu$ denoting the average true effect/outcome and $\tau^2$ denoting the variance of the true effects/outcomes in the population ($\tau^2$ is therefore often referred to as the 'amount of heterogeneity' in the true effects/outcomes). The random-effects model can therefore also be written as

$$y_i = \mu + u_i + e_i,$$

where $u_i \sim N(0, \tau^2)$ and $e_i \sim N(0, v_i)$. The fitted model provides an estimate of $\mu$ and $\tau^2$. Consequently, the random-effects model provides an *unconditional inference* about the average true effect/outcome in the population of studies (from which the $k$ studies included in the meta-analysis are assumed to be a random selection).

When including moderator variables in the random-effects model, we obtain what is typically called a **mixed-effects model** in the meta-analytic literature. Such a meta-regression model can also be

written as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + u_i + e_i,$$

where $u_i \sim N(0, \tau^2)$ and $e_i \sim N(0, v_i)$ as before, $x_{i1}$ denotes the value of the first moderator variable for the $i$th study, $x_{i2}$ denotes the value of the second moderator variable for the $i$th study, and so on (letting $p$ denote the total number of coefficients in the model, including the model intercept if it is included). Therefore, $\beta_1$ denotes how the average true effect/outcome changes for a one unit increase in $x_{i1}$, $\beta_2$ denotes how the average true effect/outcome changes for a one unit increase in $x_{i2}$, and so on, and the model intercept $\beta_0$ denotes the average true effect/outcome when the values of all moderator variables are equal to zero. The coefficients from the fitted model therefore estimate the relationship between the average true effect/outcome in the population of studies and the moderator variables included in the model. The value of $\tau^2$ in the mixed-effects model denotes the 'amount of residual heterogeneity' in the true effects/outcomes (i.e., the amount of variability in the true effects//outcomes that is not accounted for by the moderators included in the model).

When using weighted estimation in the context of a random-effects model, the model is fitted with weights equal to $w_i = 1/(\tau^2 + v_i)$, with $\tau^2$ replaced by its estimate (again, this is the standard 'inverse-variance' method for random-effects models). One can also choose unweighted estimation in the context of the random-effects model and even any user-defined weights, although the parameter that is estimated (i.e., $\mu$) remains the same regardless of the estimation method and weights used (as opposed to the fixed-effect model case, where the parameter estimated is different for weighted versus unweighted estimation or when using different weights than the standard inverse-variance weights). Since weighted estimation with inverse-variance weights is most efficient, it is usually to be preferred for random-effects models (while in the fixed-effect model case, we must carefully consider whether $\bar{\theta}_w$ or $\bar{\theta}_u$ is the more meaningful parameter to estimate). The same principle applies to mixed-effects (versus fixed-effects with moderators) models.

Contrary to what is often stated in the literature, it is important to realize that the fixed-effects model does *not* assume that the true effects/outcomes are homogeneous (i.e., that $\theta_i$ is equal to some common value $\theta$ in all $k$ studies). In other words, fixed-effects models provide perfectly valid inferences under heterogeneity, as long as one is restricting these inferences to the set of studies included in the meta-analysis and one realizes that the model does not provide an estimate of $\theta$, but of $\bar{\theta}_w$ or $\bar{\theta}_u$.

In the special case that the true effects/outcomes are actually homogeneous (the equal-effects case), the distinction between fixed- and random-effects models disappears, since homogeneity implies that $\mu = \bar{\theta}_w = \bar{\theta}_u \equiv \theta$. However, since there is no infallible method to test whether the true effects/outcomes are really homogeneous or not, a researcher should decide on the type of inference desired before examining the data and choose the model accordingly. In fact, there is nothing wrong with fitting both fixed- and random/mixed-effects models to the same data, since these models address different questions. For more details on the distinction between equal-, fixed-, and random-effects models, see Laird and Mosteller (1990) and Hedges and Vevea (1998).

**The rma.mh Function**

The Mantel-Haenszel method provides an alternative approach for fitting fixed-effects models when dealing with studies providing data in the form of $2 \times 2$ tables or in the form of event counts (i.e., person-time data) for two groups (Mantel & Haenszel, 1959). The method is particularly advantageous when aggregating a large number of studies with small sample sizes (the so-called sparse data or increasing strata case). The Mantel-Haenszel method is implemented in the `rma.mh`

function. It can be used in combination with odds ratios, relative risks, risk differences, incidence rate ratios, and incidence rate differences. The Mantel-Haenszel method is always based on a weighted estimation approach.

### The rma.peto Function

Yet another method that can be used in the context of a meta-analysis of $2 \times 2$ table data is Peto's method (see Yusuf et al., 1985), implemented in the `rma.peto` function. The method provides a weighted estimate of the (log) odds ratio under a fixed-effects model. The method is particularly advantageous when the event of interest is rare, but see the documentation of the function for some caveats.

### The rma.glmm Function

Dichotomous outcomes and event counts (based on which one can calculate effect size or outcome measures such as odds ratios, incidence rate ratios, proportions, and incidence rates) are often assumed to arise from binomial and Poisson distributed data. Meta-analytic models that are directly based on such distibutions are implemented in the `rma.glmm` function. These models are essentially special cases of generalized linear (mixed-effects) models (i.e., mixed-effects logistic and Poisson regression models). For $2 \times 2$ table data, a mixed-effects conditional logistic model (based on the non-central hypergeometric distribution) is also available. Random/mixed-effects models with dichotomous data are often referred to as 'binomial-normal' models in the meta-analytic literature. Analogously, for event count data, such models could be referred to as 'Poisson-normal' models.

### The rma.mv Function

Standard meta-analytic models assume independence between the observed effects or outcomes obtained from a set of studies. This assumption is often violated in practice. Dependencies can arise for a variety of reasons. For example, the sampling errors and/or true effects/outcomes may be correlated in multiple treatment studies (e.g., when multiple treatment groups are compared with a common control/reference group, such that the data from the control/reference group is used multiple times to compute the effect sizes or outcomes) or in multiple endpoint studies (e.g., when more than one effect size estimate or outcome is calculated based on the same sample of subjects due to the use of multiple endpoints or response variables) (Gleser & Olkin, 2009). Correlations in the true effects/outcomes can also arise due to other forms of clustering (e.g., effects/outcomes derived from the same paper, lab, research group, or species may be more similar to each other than effects/outcomes derived from different papers, labs, research groups, or species). In ecology and related fields, shared phylogenetic history among the organisms studied (e.g., plants, fungi, animals) can also induce correlations among the effects/outcomes. The `rma.mv` function can be used to fit suitable meta-analytic multivariate/multilevel models to such data, so that the non-independence in the observed/true effects or outcomes is accounted for. Network meta-analyses (also called multiple/mixed treatment comparison meta-analyses) can also be carried out with this function.

### Future Plans and Updates

The **metafor** package is a work in progress and is updated on a regular basis with new functions and options. With `metafor.news()`, you can read the 'NEWS' file of the package after installation. Comments, feedback, and suggestions for improvements are very welcome.

**Citing the Package**

To cite the package, please use the following reference:

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**Getting Started with the Package**

The paper mentioned above is a good starting place for those interested in using the **metafor** package. The purpose of the article is to provide a general overview of the package and its capabilities (as of version 1.4-0). Not all of the functions and options are described in the paper, but it should provide a useful introduction to the package. The paper can be freely downloaded from the URL given above or can be directly loaded with the command vignette("metafor").

In addition to reading the paper, carefully read this page and then the help pages for the escalc and the rma.uni functions (or the rma.mh, rma.peto, rma.glmm, rma.mv functions if you intend to use these methods). The help pages for these functions provide links to many additional functions, which can be used after fitting a model.

A (pdf) diagram showing the various functions in the metafor package (and how they are related to each other) can be opened with the command vignette("metafor_diagram").

Finally, additional information about the package, several detailed analysis examples, examples of plots and figures provided by the package (with the corresponding code), some additional tips and notes, and a FAQ can be found on the package website at http://www.metafor-project.org/.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

Suggestions on how to obtain help with using the package can found on the package website at:
http://www.metafor-project.org/doku.php/help

**References**

Cooper, H., Hedges, L. V., & Valentine, J. C. (Eds.) (2009). *The handbook of research synthesis and meta-analysis* (2nd ed.). New York: Russell Sage Foundation.

Gleser, L. J., & Olkin, I. (2009). Stochastically dependent effect sizes. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 357–376). New York: Russell Sage Foundation.

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. San Diego, CA: Academic Press.

Hedges, L. V., & Vevea, J. L. (1998). Fixed- and random-effects models in meta-analysis. *Psychological Methods*, **3**, 486–504.

Laird, N. M., & Mosteller, F. (1990). Some statistical methods for combining experimental results. *International Journal of Technology Assessment in Health Care*, **6**, 5–30.

Mantel, N., & Haenszel, W. (1959). Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute*, **22**, 719–748.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Yusuf, S., Peto, R., Lewis, J., Collins, R., & Sleight, P. (1985). Beta blockade during and after myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Disease*, **27**, 335–371.

---

addpoly                          *Add Polygons to Forest Plots*

---

### Description

The function addpoly is generic. It can be used to add polygons to a forest plot, for example, to indicate summary estimates for all or subgroups of studies and to indicate fitted/predicted values based on models involving moderators.

### Usage

```
addpoly(x, ...)
```

### Arguments

x              either an object of class "rma" or the values at which polygons should be drawn.
               See 'Details'.

...            other arguments.

### Details

Currently, methods exist for two types of situations.

In the first case, object x is a fitted model coming from the rma.uni, rma.mh, rma.peto, or rma.glmm functions. The model must either be a fixed- or random-effects model, that is, the model should not contain any moderators. The corresponding method is called addpoly.rma. It can be used to add a polygon to an existing forest plot (often at the bottom), showing the summary estimate with corresponding confidence interval based on the fitted model.

Alternatively, object x can be a vector with values at which one or more polygons should be drawn. The corresponding method is then addpoly.default.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

addpoly.rma, addpoly.default, forest.rma, forest.default

---

addpoly.default                    *Add Polygons to Forest Plots (Default Method)*

---

**Description**

Function to add one or more polygons to a forest plot.

**Usage**

```
## Default S3 method:
addpoly(x, vi, sei, ci.lb, ci.ub, rows=-1, level=95, digits=2,
        annotate=TRUE, mlab, transf, atransf, targs,
        efac=1, col, border, cex, ...)
```

**Arguments**

| | |
|---|---|
| x | vector with the values at which the polygons should be drawn. |
| vi | vector with the corresponding variances. |
| sei | vector with the corresponding standard errors.<br>(note: only one of the two, vi or sei, needs to be specified) |
| ci.lb | vector with the corresponding lower confidence intervals bounds. Not needed if vi or sei is specified. See 'Details'. |
| ci.ub | vector with the corresponding upper confidence intervals bounds. Not needed if vi or sei is specified. See 'Details'. |
| rows | vector specifying the rows (or more generally, the horizontal positions) for plotting the polygons (defaults is -1). Can also be a single value specifying the row (horizontal position) of the first polygon (the remaining polygons are then plotted below this starting row). |
| level | numerical value between 0 and 100 specifying the confidence interval level (default is 95). |
| digits | integer specifying the number of decimal places to which the annotations should be rounded (default is 2). |
| annotate | logical specifying whether annotations should be added to the plot for the polygons that are drawn (default is TRUE). |
| mlab | optional character vector with the same length as x giving labels for the polygons that are drawn. |
| transf | optional argument specifying the name of a function that should be used to transform the x values and confidence interval bounds. If unspecified, no transformation is used. |
| atransf | optional argument specifying the name of a function that should be used to transform the annotations. If unspecified, no transformation is used. |

| targs | optional arguments needed by the function specified via `transf` or `atransf`. |
|---|---|
| efac | vertical expansion factor for the polygons. The default value of 1 should usually work okay. |
| col | optional character string specifying the name of a color to use for the polygons. If unspecified, the function sets a default color. |
| border | optional character string specifying the name of a color to use for the border of the polygons. If unspecified, the function sets a default color. |
| cex | optional symbol expansion factor. If unspecified, the function tries to set this to a sensible value. |
| ... | other arguments. |

## Details

The function can be used to add one or more polygons to an existing forest plot created with the [forest](#) function. For example, summary estimates based on a model involving moderators can be added to the plot this way. See example below.

To use the function, one should specify the values at which the polygons should be drawn (via the x argument) together with the corresponding variances (via the vi argument) or with the corresponding standard errors (via the sei argument). Alternatively, one can specify the values at which the polygons should be drawn together with the corresponding confidence interval bounds (via the ci.lb and ci.ub arguments).

The arguments transf, atransf, efac, and cex should always be set equal to the same values used to create the forest plot.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

[forest.rma](#), [forest.default](#)

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-
### effects model with absolute latitude as a moderator
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat,
           slab=paste(author, year, sep=", "), data=dat.bcg)
```

```
### forest plot of the observed relative risks
forest(res, addfit=FALSE, atransf=exp, xlim=c(-8,5), ylim=c(-4.5,16), cex=.8,
       order=order(dat.bcg$ablat), ilab=dat.bcg$ablat, ilab.xpos=-3.5)

### predicted average log relative risk for 10, 30, and 50 degrees absolute latitude
x <- predict(res, newmods=c(10, 30, 50))

### add predicted average relative risks to forest plot
addpoly(x$pred, sei=x$se, atransf=exp, col="white", rows=-2,
        mlab=c("- at 10 Degrees", "- at 30 Degrees", "- at 50 Degrees"), cex=.8)
abline(h=0)
text(-8,   -1, "Model-Based Estimates of RR:", pos=4, cex=.8)
text(-8,   15, "Author(s) and Year",      pos=4, cex=.8)
text(5,    15, "Relative Risk [95% CI]", pos=2, cex=.8)
text(-3.5, 15, "Latitude", cex=.8)
```

---

addpoly.rma                           *Add Polygons to Forest Plots (Method for 'rma' Objects)*

---

### Description

Function to add a polygon to a forest plot showing the summary estimate with correspondong confidence interval based on an object of class "rma".

### Usage

```
## S3 method for class 'rma'
addpoly(x, row=-2, level=x$level, digits=2,
        annotate=TRUE, mlab, transf, atransf, targs,
        efac=1, col, border, cex, ...)
```

### Arguments

| | |
|---|---|
| x | an object of class "rma". |
| row | value specifying the row (or more generally, the horizontal position) for plotting the polygon (default is -2). |
| level | numerical value between 0 and 100 specifying the confidence interval level (the default is to take the value from the object). |
| digits | integer specifying the number of decimal places to which the annotations should be rounded (default is 2). |
| annotate | logical specifying whether annotations for the summary estimate should be added to the plot (default is TRUE). |
| mlab | optional character string giving a label for the summary estimate polygon. If unspecified, the function sets a default label. |

| transf | optional argument specifying the name of a function that should be used to transform the summary estimate and confidence interval bound. If unspecified, no transformation is used. |
|---|---|
| atransf | optional argument specifying the name of a function that should be used to transform the annotations. If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified via transf or atransf. |
| efac | vertical expansion factor for the polygon. The default value of 1 should usually work okay. |
| col | optional character string specifying the name of a color to use for the polygon. If unspecified, the function sets a default color. |
| border | optional character string specifying the name of a color to use for the border of the polygon. If unspecified, the function sets a default color. |
| cex | optional symbol expansion factor. If unspecified, the function tries to set this to a sensible value. |
| ... | other arguments. |

## Details

The function can be used to add a polygon to an existing forest plot created with the [forest](#) function. The polygon shows the summary estimate based on a fixed- or random-effects model. Using this function, summary estimates based on different types of models can be shown in the same plot. Also, summary estimates based on a subgrouping of the studies can be added to the plot this way. See examples below.

The arguments transf, atransf, efac, and cex should always be set equal to the same values used to create the forest plot.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[forest.rma](#), [forest.default](#)

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using the Mantel-Haenszel method
```

```
res <- rma.mh(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
              slab=paste(author, year, sep=", "))

### forest plot of the observed relative risks with summary estimate
forest(res, atransf=exp, xlim=c(-7,5), ylim=c(-2.5,16))

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### add summary estimate from the random-effects model to forest plot
addpoly(res, atransf=exp)

### forest plot with subgrouping of studies and summaries per subgroup
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
           slab=paste(author, year, sep=", "))
forest(res, xlim=c(-16, 6), at=log(c(.05, .25, 1, 4)), atransf=exp,
       ilab=cbind(dat.bcg$tpos, dat.bcg$tneg, dat.bcg$cpos, dat.bcg$cneg),
       ilab.xpos=c(-9.5,-8,-6,-4.5), cex=.75, ylim=c(-1, 27),
       order=order(dat.bcg$alloc), rows=c(3:4,9:15,20:23),
       mlab="RE Model for All Studies")
op <- par(cex=.75, font=4)
text(-16, c(24,16,5), c("Systematic Allocation", "Random Allocation",
                        "Alternate Allocation"), pos=4)
par(font=2)
text(c(-9.5,-8,-6,-4.5), 26, c("TB+", "TB-", "TB+", "TB-"))
text(c(-8.75,-5.25),     27, c("Vaccinated", "Control"))
text(-16,                26, "Author(s) and Year",    pos=4)
text(6,                  26, "Relative Risk [95% CI]", pos=2)
par(op)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
           subset=(alloc=="systematic"))
addpoly(res, row=18.5, cex=.75, atransf=exp, mlab="RE Model for Subgroup")
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
           subset=(alloc=="random"))
addpoly(res, row=7.5, cex=.75, atransf=exp, mlab="RE Model for Subgroup")
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
           subset=(alloc=="alternate"))
addpoly(res, row=1.5, cex=.75, atransf=exp, mlab="RE Model for Subgroup")
```

---

anova.rma                           *Likelihood Ratio and Wald-Type Tests for 'rma' Objects*

---

**Description**

For two (nested) models of class "rma.uni" or "rma.mv", the function provides a full versus reduced model comparison in terms of model fit statistics and a likelihood ratio test. When a single model is specified, a Wald-type test of one or more model coefficients is carried out.

## Usage

```
## S3 method for class 'rma'
anova(object, object2, btt, digits, ...)
```

## Arguments

| | |
|---|---|
| object | an object of class "rma.uni" or "rma.mv". |
| object2 | an (optional) object of class "rma.uni" or "rma.mv". See 'Details'. |
| btt | optional vector of indices specifying which coefficients to include in the Wald-type test. See 'Details'. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

## Details

When a single model is specified, the function provides a Wald-type test of one or more model coefficients. For a fixed- or random-effects model (i.e., a model without moderators), this is just the test of the single coefficient of the model. For models including moderators, an omnibus test of all the model coefficients is conducted that excludes the intercept (the first coefficient) if it is included in the model. If no intercept is included in the model, then the omnibus test includes all of the coefficients in the model including the first. Alternatively, one can manually specify the indices of the coefficients to test via the btt argument. For example, with btt=c(3,4), only the third and fourth coefficient from the model would be included in the test (if an intercept is included in the model, then it corresponds to the first coefficient in the model).

When specifying two models for comparison, the function provides a likelihood ratio test comparing the two models. The two models must be based on the same set of data, must be of the same class, and should be nested for the likelihood ratio test to make sense.

## Value

An object of class "anova.rma". When a single model is specified, the object is a list containing the following components:

| | |
|---|---|
| QM | test statistic for the Wald-type test of model coefficients. |
| QMp | p-value for the Wald-type test of model coefficients. |
| btt | indices of the coefficients tested by the Wald-type test. |
| k | number of outcomes included in the model fitting. |
| p | number of coefficients in the model (including the intercept). |
| m | number of coefficients included in the Wald-type test of coefficients. |
| ... | some additional elements/values. |

When two models are specified, the object is a list containing the following components:

| | |
|---|---|
| fit.stats.f | log-likelihood, deviance, AIC, BIC, and AICc for the full model. |

| | |
|---|---|
| `fit.stats.r` | log-likelihood, deviance, AIC, BIC, and AICc for the reduced model. |
| `p.f` | number of parameters in the full model. |
| `p.r` | number of parameters in the reduced model. |
| `LRT` | likelihood ratio test statistic. |
| `pval` | p-value for the likelihood ratio test. |
| `QE.f` | test statistic for the test of (residual) heterogeneity from the full model. |
| `QE.r` | test statistic for the test of (residual) heterogeneity from the reduced model. |
| `tau2.f` | estimated $\tau^2$ value from the full model. `NA` for `"rma.mv"` objects. |
| `tau2.r` | estimated $\tau^2$ value from the reduced model. `NA` for `"rma.mv"` objects. |
| `R2` | amount of (residual) heterogeneity in the reduced model that is accounted for in the full model. `NA` for fixed-effects models, if the amount of heterogeneity in the reduced model is equal to zero, or for `"rma.mv"` objects. This can be regarded as a pseudo $R^2$ statistic (Raudenbush, 2009). |
| `...` | some additional elements/values. |

The results are formated and printed with the `print.anova.rma` function.

## Note

Note that likelihood ratio tests are not meaningful when using REML estimation and the two models have different fixed effects.

In principle, one can also consider likelihood ratio tests for amount of (residual) heterogeneity in random- and mixed-effects models. The full model should then be fitted with either `method="ML"` or `method="REML"` and the reduced model with `method="FE"`. The p-value from that test is based on a chi-square distribution with 1 degree of freedom, but actually needs to be adjusted for the fact that the parameter (i.e., $\tau^2$) falls on the boundary of the parameter space under the null hypothesis. Moreover, the Q-test usually keeps better control of the Type I error rate and therefore should be preferred (see Viechtbauer, 2007, for more details).

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Hardy, R. J., & Thompson, S. G. (1996). A likelihood approach to meta-analysis with random effects. *Statistics in Medicine*, **15**, 619–629.

Huizenga, H. M., Visser, I., & Dolan, C. V. (2011). Testing overall and moderator effects in random effects meta-regression. *British Journal of Mathematical and Statistical Psychology*, **64**, 1–19.

Raudenbush, S. W. (2009). Analyzing effect sizes: Random effects models. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 295–315). New York: Russell Sage Foundation.

Viechtbauer, W. (2007). Hypothesis tests for population heterogeneity in meta-analysis. *British Journal of Mathematical and Statistical Psychology*, **60**, 29–60.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

### See Also

rma.uni, print.anova.rma

### Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res2 <- rma(yi, vi, data=dat, method="ML")

### mixed-effects model with two moderators (absolute latitude and publication year)
res1 <- rma(yi, vi, mods = ~ ablat + year, data=dat, method="ML")

### Wald-type test of the two moderators
anova(res1)

### corresponding likelihood ratio test
anova(res1, res2)
```

---

baujat                          *Baujat Plots for 'rma' Objects*

---

### Description

Function to create Baujat plots for objects of class "rma.uni", "rma.mh", and "rma.peto".

### Usage

```
baujat(x, ...)

## S3 method for class 'rma.uni'
baujat(x, xlim, ylim, xlab, ylab, cex, grid=TRUE, ...)

## S3 method for class 'rma.mh'
baujat(x, xlim, ylim, xlab, ylab, cex, grid=TRUE, ...)

## S3 method for class 'rma.peto'
baujat(x, xlim, ylim, xlab, ylab, cex, grid=TRUE, ...)
```

**Arguments**

| | |
|---|---|
| x | an object of class "rma.uni", "rma.mh", or "rma.peto". |
| xlim | x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| ylim | y-axis limits. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| ylab | title for the y-axis. If unspecified, the function tries to set an appropriate axis title. |
| cex | optional character expansion factor. If unspecified, the function tries to set this to a sensible value. |
| grid | logical indicating whether a grid should be added to the plot. |
| ... | other arguments. |

**Details**

Baujat et al. (2002) proposed a diagnostic plot to detect sources of heterogeneity in meta-analytic data. The plot shows the contribution of each study to the overall Q-test statistic for heterogeneity on the x-axis versus the influence of each study (defined as the standardized squared difference between the overall estimate based on a fixed-effects model with and without the study included in the model fitting) on the y-axis. The same type of plot can be produced by first fitting a fixed-effects model with either the rma.uni (using method="FE"), rma.mh, or rma.peto functions and then passing the fitted model object to the baujat function.

For models fitted with the rma.uni function (which may involve moderators and/or may be random/mixed-effects models), the idea underlying this type of plot can be generalized as follows: The x-axis then corresponds to the squared Pearson residual of a study, while the y-axis corresponds to the standardized squared difference between the predicted/fitted value for the study with and without the study included in the model fitting. Therefore, for a fixed-effect with moderators model, the x-axis corresponds to the contribution of the study to the QE-test statistic for residual heterogeneity.

The points are labeled according to the slab argument (see model fitting functions). If slab was unspecified, then the numbers 1 through $k$ are used by default.

**Value**

A data frame with components:

| | |
|---|---|
| x | the x coordinates of the points that were plotted. |
| y | the y coordinates of the points that were plotted. |
| slab | study labels of the points that were plotted. |

Note that the data frame is returned invisibly.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Baujat, B., Mahe, C., Pignon, J.-P., & Hill, C. (2002). A graphical method for exploring heterogeneity in meta-analyses: Application to a meta-analysis of 65 trials. *Statistics in Medicine*, **21**(18), 2641–2652.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.mh, rma.peto, influence.rma.uni, radial

## Examples

```
### load data from Pignon et al. (2000)
dat <- get(data(dat.pignon2000))

### compute estimated log hazard ratios and sampling variances
dat$yi <- with(dat, OmE/V)
dat$vi <- with(dat, 1/V)

### meta-analysis based on all 65 trials
res <- rma(yi, vi, data=dat, method="FE", slab=id)

### create Baujat plot
baujat(res)
```

---

bldiag                          *Construct Block Diagonal Matrix*

---

## Description

Function to construct a block diagonal matrix from (a list of) matrices.

## Usage

```
bldiag(...)
```

## Arguments

...          individual matrices or a list of matrices.

## Author(s)

Posted to R-help by Berton Gunter (2 Sep 2005).

## See Also

[rma.mv](#)

## Examples

```
### load data
dat <- get(data(dat.berkey1998))

### construct list of the variance-covariance matrices of the observed outcomes for the studies
V <- lapply(split(dat[,c("v1i", "v2i")], dat$trial), as.matrix)

### construct block diagonal matrix
V <- bldiag(V)
V
```

---

blup                        *Best Linear Unbiased Predictions for 'rma.uni' Objects*

---

## Description

The function calculates best linear unbiased predictions (BLUPs) of the true outcomes by combining the fitted values based on the fixed effects and the estimated contributions of the random effects for objects of class `"rma.uni"`. Corresponding standard errors and prediction interval bounds are also provided.

## Usage

```
blup(x, ...)

## S3 method for class 'rma.uni'
blup(x, level, digits, transf, targs, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"rma.uni"`. |
| level | numerical value between 0 and 100 specifying the prediction interval level (if unspecified, the default is to take the value from the object). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| transf | optional argument specifying the name of a function that should be used to transform the predicted values and interval bounds (e.g., `transf=exp`). If unspecified, no transformation is used. |

targs            optional arguments needed by the function specified under `transf`.

...               other arguments.

## Value

An object of class `"list.rma"`. The object is a list containing the following components:

pred            predicted values.

se              corresponding standard errors.

pi.lb           lower bound of the prediction intervals.

pi.ub           upper bound of the prediction intervals.

...               some additional elements/values.

The `"list.rma"` object is formated and printed with `print.list.rma`.

## Note

For predicted/fitted values that are based only on the fixed effects of the model, see `fitted.rma` and `predict.rma`.

Fixed-effects models (with or without moderators) do not contain random study effects. The BLUPs for these models will therefore automatically be equal to the usual fitted values, that is, those obtained with `fitted.rma` and `predict.rma`.

When using the `transf` argument, the transformation is applied to the predicted values and the corresponding interval bounds. The standard errors are set equal to `NA`.

The normal distribution is used to calculate the prediction intervals. When the model was fitted with the Knapp and Hartung (2003) method (i.e., `knha=TRUE` in the `rma.uni` function), then the t-distribution with $k - p$ degrees of freedom is used.

To be precise, it should be noted that the function actually calculates empirical BLUPs (eBLUPs), since the predicted values are a function of the estimated value of $\tau^2$.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Kackar, R. N., & Harville, D. A. (1981). Unbiasedness of two-stage estimation and prediction procedures for mixed linear models. Communications in Statistics, Theory and Methods, **10**, 1249–1261.

Raudenbush, S. W., & Bryk, A. S. (1985). Empirical Bayes meta-analysis. *Journal of Educational Statistics*, **10**, 75–98.

Robinson, G. K. (1991). That BLUP is a good thing: The estimation of random effects. *Statistical Science*, **6**, 15–32.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

[rma.uni](), [predict.rma](), [fitted.rma]()

**Examples**

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(yi, vi, data=dat)
blup(res, transf=exp)

### illustrate shrinkage of BLUPs towards the (estimated) population average
res <- rma(yi, vi, data=dat)
blups <- blup(res)$pred
plot(NA, NA, xlim=c(.8,2.4), ylim=c(-2,0.5), pch=19,
     xaxt="n", bty="n", xlab="", ylab="Log Relative Risk")
segments(rep(1,13), dat$yi, rep(2,13), blups, col="darkgray")
points(rep(1,13), dat$yi, pch=19)
points(rep(2,13), blups, pch=19)
axis(side=1, at=c(1,2), labels=c("Observed\nValues", "BLUPs"), lwd=0)
segments(.7, res$b, 2.15, res$b, lty="dotted")
text(2.3, res$b, expression(hat(mu)==-0.71), cex=1)
```

---

coef.permutest.rma.uni

*Extract the Model Coefficient Table from 'permutest.rma.uni' Objects*

---

**Description**

The function extracts the estimated model coefficients, corresponding standard errors, test statistics, p-values (based on the permutation test), and confidence interval bounds from objects of class "permutest.rma.uni".

**Usage**

```
## S3 method for class 'permutest.rma.uni'
coef(object, ...)
```

**Arguments**

| | |
|---|---|
| object | an object of class "permutest.rma.uni". |
| ... | other arguments. |

## Value

A data frame with the following elements:

| | |
|---|---|
| estimate | estimated model coefficient(s). |
| se | corresponding standard error(s). |
| zval | corresponding test statistic(s). |
| pval | p-value(s) based on the permutation test. |
| ci.lb | lower bound of the confidence interval(s). |
| ci.ub | upper bound of the confidence interval(s). |

When the model was fitted with the Knapp and Hartung (2003) method (i.e., knha=TRUE in the rma.uni function), then zval is called tval in the data frame that is returned by the function.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

permutest.rma.uni, rma.uni

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)

### permutation test
## Not run: pres <- permutest(res)
## Not run: coef(pres)
```

| coef.rma | *Extract the Model Coefficients and Coefficient Table from 'rma' and 'summary.rma' Objects* |
|---|---|

## Description

The coef function extracts the estimated model coefficients from objects of class `"rma"`. For objects of class `"summary.rma"`, the model coefficients, corresponding standard errors, test statistics, p-values, and confidence interval bounds are extracted.

## Usage

```
## S3 method for class 'rma'
coef(object, ...)

## S3 method for class 'summary.rma'
coef(object, ...)
```

## Arguments

| | |
|---|---|
| object | an object of class `"rma"` or `"summary.rma"`. |
| ... | other arguments. |

## Value

Either a vector with the estimated model coefficient(s) or a data frame with the following elements:

| | |
|---|---|
| estimate | estimated model coefficient(s). |
| se | corresponding standard error(s). |
| zval | corresponding test statistic(s). |
| pval | corresponding p-value(s). |
| ci.lb | corresponding lower bound of the confidence interval(s). |
| ci.ub | corresponding upper bound of the confidence interval(s). |

When the model was fitted with the Knapp and Hartung (2003) method (i.e., knha=TRUE in the [rma.uni](#) function), then zval is called tval in the data frame that is returned by the function.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

### See Also

[rma.uni](), [rma.mh](), [rma.peto](), [rma.glmm](), [rma.mv]()

### Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
coef(res)
coef(summary(res))
```

---

| confint.rma.uni | *Confidence Intervals for 'rma' Objects* |
|---|---|

---

### Description

The function calculates a confidence interval for the amount of (residual) heterogeneity in random- and mixed-effects models that were fitted with the [rma.uni]() function. Confidence intervals for the model coefficients can also be obtained.

### Usage

```
## S3 method for class 'rma.uni'
confint(object, parm, level, fixed=FALSE, random=TRUE,
        digits, verbose=FALSE, control, ...)

## S3 method for class 'rma.mh'
confint(object, parm, level, digits, ...)

## S3 method for class 'rma.peto'
confint(object, parm, level, digits, ...)

## S3 method for class 'rma.glmm'
confint(object, parm, level, digits, ...)

## S3 method for class 'rma.mv'
confint(object, parm, level, digits, ...)
```

### Arguments

| | |
|---|---|
| object | an object of class "rma.uni", "rma.mh", or "rma.peto". The method is not yet implemented for objects of class "rma.glmm" or "rma.mv". |
| parm | this argument is here for compatability with the generic function [confint](), but is ignored. |

| fixed | logical indicating whether confidence intervals for the model coefficients should be returned (default is FALSE). |
|---|---|
| random | logical indicating whether a confidence interval for the amount of (residual) heterogeneity should be returned (default is TRUE). |
| level | numerical value between 0 and 100 specifying the confidence interval level (if unspecified, the default is to take the value from the object). |
| digits | integer specifying the number of decimal places to which the results should be rounded (if unspecified, the default is to take the value from the object). |
| verbose | logical indicating whether output should be generated on the progress of the iterative algorithms used as part of the Q-profile and generalized Q-statistic methods (default is FALSE). See 'Note'. |
| control | list of control values for the iterative algorithm. If unspecified, default values are defined inside the function. See 'Note'. |
| ... | other arguments. |

## Details

The confidence intervals for the model coefficients are simply the usual Wald-type intervals which are also shown when printing the fitted object.

The confidence interval for the amount of (residual) heterogeneity is obtained iteratively either via the Q-profile method as described by Hartung and Knapp (2005) and Viechtbauer (2007) or via the generalized Q-statistic method described by Jackson (2013). The latter is automatically used when the model was fitted with method="GENQ", the former is used in all other cases. Either method provides an exact confidence interval for $\tau^2$ in random- and mixed-effects models. The square root of the interval bounds is also returned for easier interpretation. Confidence intervals for $I^2$ and $H^2$ are also provided (Higgins & Thompson, 2002). Since $I^2$ and $H^2$ are just monotonic transformation of $\tau^2$, the confidence intervals for $I^2$ and $H^2$ are also exact.

## Value

An object of class "confint.rma". The object is a list with either one or two elements (named fixed and random) with the following elements:

| estimate | estimate of the model coefficient or variance component. |
|---|---|
| ci.lb | lower bound of the confidence interval. |
| ci.ub | upper bound of the confidence interval. |

For fixed-effects models, the variance component estimates are NA, but the confidence interval bounds are still provided (but they are essentially meaningless for fixed-effects models).

The results are formated and printed with the [print.confint.rma](#) function.

## Note

The iterative algorithm used as part of the Q-profile and generalized Q-statistic methods makes use of the [uniroot](#) function. By default, the desired accuracy is set equal to .Machine$double.eps^0.25 and the maximum number of iterations to 1000. The upper bound of the interval searched is set to 100 (which should be large enough for most cases). The desired accuracy (tol), maximum number

of iterations (`maxiter`), and upper bound (`tau2.max`) can be adjusted with `control=list(tol=value, maxiter=value, ta`
For the Q-profile method, one can also adjust the lower bound of the interval searched with `control=list(tau2.min=value)`
(the default is to take that value from the object, which is 0 by default). You should only play
around with this value if you know what you are doing. If `verbose=TRUE`, output is generated on
the progress of the iterative algorithm.

It is possible that the lower and upper confidence interval bounds both fall below zero (or whatever
value was chosen for `tau2.min`). Since both values then fall outside of the parameter space, the
confidence interval then just consists of the null/empty set. Alternatively, one could interpret this as
a CI with bounds $[0, 0]$ or as indicating 'highly homogeneous' data.

Usually, the estimate of $\tau^2$ from the random/mixed-effects model will fall within the confidence in-
terval provided by the Q-profile method. However, this is not guaranteed. Depending on the method
used to estimate $\tau^2$ and the width of the confidence interval, it can happen that the confidence inter-
val does not actually contain the estimate (trying to explain this to reviewers can be tricky). How-
ever, using the empirical Bayes estimator of $\tau^2$ when fitting the model (i.e., using `method="EB"`)
guarantees that the estimate of $\tau^2$ falls within the confidence interval. When `method="GENQ"` was
used to fit the model, the corresponding CI obtained via the generalized Q-statistic method is also
guaranteed to contain the estimate $\tau^2$.

The Q-profile and generalized Q-statistic methods are both exact under the assumptions of the
random- and mixed-effects models (i.e., normally distributed observed and true outcomes and
known sampling variances). In practice, these assumptions are usually only approximately true,
turning confidence intervals for $\tau^2$ also into approximations.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Hartung, J., & Knapp, G. (2005). On confidence intervals for the among-group variance in the
one-way random effects model with unequal error variances. *Journal of Statistical Planning and
Inference*, **127**, 157–177.

Higgins, J. P. T., & Thompson, S. G. (2002). Quantifying heterogeneity in a meta-analysis. *Statis-
tics in Medicine*, **21**, 1539–1558.

Jackson, D. (2013). Confidence intervals for the between-study variance in random effects meta-
analysis using generalised Cochran heterogeneity statistics. *Research Synthesis Methods*, **4**, 220–
229.

Mantel, N., & Haenszel, W. (1959). Statistical aspects of the analysis of data from retrospective
studies of disease. *Journal of the National Cancer Institute*, **22**, 719–748.

Viechtbauer, W. (2007). Confidence intervals for the amount of heterogeneity in meta-analysis.
*Statistics in Medicine*, **26**, 37–52.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of
Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Yusuf, S., Peto, R., Lewis, J., Collins, R., & Sleight, P. (1985). Beta blockade during and after
myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Disease*,
**27**, 335–371.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### confidence interval for the total amount of heterogeneity
confint(res)

### mixed-effects model with absolute latitude in the model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat, data=dat.bcg)

### confidence interval for the residual amount of heterogeneity
confint(res)
```

---

cumul                        *Cumulative Meta-Analysis for 'rma' Objects*

---

## Description

The functions repeatedly fit the specified model, adding one observation/study at a time to the
model.

## Usage

```
cumul(x, ...)

## S3 method for class 'rma.uni'
cumul(x, order, digits, transf, targs, ...)
## S3 method for class 'rma.mh'
cumul(x, order, digits, transf, targs, ...)
## S3 method for class 'rma.peto'
cumul(x, order, digits, transf, targs, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "rma.mh", "rma.peto", "rma.uni". |
| order | optional vector with indices giving the desired order for the cumulative meta-analysis. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |

| transf | optional argument specifying the name of a function that should be used to transform the model coefficients and interval bounds (e.g., `transf=exp`). If unspecified, no transformation is used. |
|---|---|
| targs | optional arguments needed by the function specified under `transf`. |
| ... | other arguments. |

## Details

For `"rma.uni"` objects, the model specified by `x` must be a model without moderators (i.e., either a fixed- or a random-effects model).

## Value

An object of class `c("list.rma","cumul.rma")`. The object is a list containing the following components:

| estimate | estimated coefficients of the model. |
|---|---|
| se | standard errors of the coefficients. |
| zval | test statistics of the coefficients. |
| pval | p-values for the test statistics. |
| ci.lb | lower bounds of the confidence intervals for the coefficients. |
| ci.ub | upper bounds of the confidence intervals for the coefficients. |
| QE | test statistics for the tests of heterogeneity. |
| QEp | p-values for the tests of heterogeneity. |
| tau2 | estimated amounts of (residual) heterogeneity (only for random-effects models). |
| I2 | values of $I^2$ (only for random-effects models). |
| H2 | values of $H^2$ (only for random-effects models). |

The object is formated and printed with [print.list.rma](). A forest plot showing the results from the cumulative meta-analysis can be obtained with [forest.cumul.rma](). For random-effects models, [plot.cumul.rma]() can also be used to visualize the results.

## Note

When using the `transf` option, the transformation is applied to the estimated coefficients and the corresponding interval bounds. The standard errors are then set equal to `NA` and are omitted from the printed output.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Chalmers, T. C., & Lau, J. (1993). Meta-analytic stimulus for changes in clinical trials. *Statistical Methods in Medical Research*, **2**, 161–172.

Lau, J., Schmid, C. H., & Chalmers, T. C. (1995). Cumulative meta-analysis of clinical trials builds evidence for exemplary medical care. *Journal of Clinical Epidemiology*, **48**, 45–57.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

forest.cumul.rma, plot.cumul.rma

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res <- rma(yi, vi, data=dat)

### cumulative meta-analysis (in the order of publication year)
cumul(res, transf=exp, order=order(dat$year))

### meta-analysis of the (log) relative risks using the Mantel-Haenszel method
res <- rma.mh(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### cumulative meta-analysis
cumul(res, order=order(dat.bcg$year))
cumul(res, order=order(dat.bcg$year), transf=TRUE)

### meta-analysis of the (log) odds ratios using Peto's method
res <- rma.mh(ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### cumulative meta-analysis
cumul(res, order=order(dat.bcg$year))
cumul(res, order=order(dat.bcg$year), transf=TRUE)
```

---

dat.bangertdrowns2004    *Studies on the Effectiveness of Writing-to-Learn Interventions*

---

## Description

Results from 48 studies on the effectiveness of school-based writing-to-learn interventions on academic achievement.

**Usage**

```
dat.bangertdrowns2004
```

**Format**

The data frame contains the following columns:

| | | |
|---|---|---|
| **id** | numeric | study number |
| **author** | character | study author(s) |
| **year** | numeric | publication year |
| **grade** | numeric | grade level (1 = elementary; 2 = middle; 3 = high-school; 4 = college) |
| **length** | numeric | treatment length (in weeks) |
| **minutes** | numeric | minutes per assignment |
| **wic** | numeric | writing in class (0 = no; 1 = yes) |
| **feedback** | numeric | feedback (0 = no; 1 = yes) |
| **info** | numeric | writing contained informational components (0 = no; 1 = yes) |
| **pers** | numeric | writing contained personal components (0 = no; 1 = yes) |
| **imag** | numeric | writing contained imaginative components (0 = no; 1 = yes) |
| **meta** | numeric | prompts for metacognitive reflection (0 = no; 1 = yes) |
| **subject** | character | subject matter |
| **ni** | numeric | total sample size of the study |
| **yi** | numeric | standardized mean difference |
| **vi** | numeric | corresponding sampling variance |

**Details**

In each of the studies included in this meta-analysis, an experimental group (i.e., a group of students that received instruction with increased emphasis on writing tasks) was compared against a control group (i.e., a group of students that received conventional instruction) with respect to some content-related measure of academic achievement (e.g., final grade, an exam/quiz/test score). The effect size measure for this meta-analysis was the standardized mean difference (with positive scores indicating a higher mean level of academic achievement in the intervention group).

The standardized mean differences given here are bias-corrected and therefore differ slighty from the values reported in the article. Also, since only the total sample size is given in the article, the estimated sampling variances were computed under the assumption that $n_{i1} = n_{i2} = n_i/2$.

**Source**

Bangert-Drowns, R. L., Hurley, M. M., & Wilkinson, B. (2004). The effects of school-based writing-to-learn interventions on academic achievement: A meta-analysis. *Review of Educational Research*, **74**, 29–58.

**Examples**

```
### load data
dat <- get(data(dat.bangertdrowns2004))

### random-effects model
res <- rma(yi, vi, data=dat)
```

```
res
```

---

| dat.begg1989 | *Studies on Bone-Marrow Transplantation versus Chemotherapy for the Treatment of Leukemia* |
| --- | --- |

---

#### Description

Results from controlled and uncontrolled studies on the effectiveness of allogeneic bone-marrow transplantation (BMT) and conventional chemotherapy (CMO) in the treatment of acute nonlymphocytic leukemia.

#### Usage

```
dat.berkey1989
```

#### Format

The data frame contains the following columns:

| | | |
| --- | --- | --- |
| **study** | numeric | study number |
| **trt** | character | treatment (BMT or CMO) |
| **arms** | numeric | number of arms in the study (1 = uncontrolled studies; 2 = controlled studies) |
| **yi** | numeric | 2-year disease-free survival |
| **sei** | numeric | standard error |
| **v2i** | numeric | sampling variance |

#### Details

The dataset includes the results from controlled and uncontrolled studies on the 2-year disease-free survival in patients with acute nonlymphocytic leukemia receiving either allogeneic bone-marrow transplantation (BMT) or conventional chemotherapy (CMO). In the controlled (two-arm) studies, a cohort of patients in complete remission and potentially eligible for BMT was assembled, and those who consented and for whom a donor could be found received BMT, with the remaining patients used as controls (receiving CMO). In the uncontrolled (one-arm) studies, only a single group was studied, receiving either BMT or CMO.

The data in this dataset were obtained from Table 1 in Begg & Pilote (1991, p. 902).

#### Source

Begg, C. B., & Pilote, L. (1991). A model for incorporating historical controls into a meta-analysis. *Biometrics*, **47**, 899–906.

#### References

Begg, C. B., Pilote, L., & McGlave, P. B. (1989). Bone marrow transplantation versus chemotherapy in acute non-lymphocytic leukemia: A meta-analytic review. *European Journal of Cancer and Clinical Oncology*, **25**, 1519–1523.

## Examples

```
### load data
dat <- get(data(dat.begg1989))

### turn trt and arms into factors and set reference levels
dat$trt  <- relevel(factor(dat$trt), ref="CMO")
dat$arms <- relevel(factor(dat$arms), ref="2")

### create data frame with the treatment differences for the controlled studies
dat2 <- data.frame(yi = dat$yi[c(1,3,5,7)] - dat$yi[c(2,4,6,8)],
                   vi = dat$vi[c(1,3,5,7)] + dat$vi[c(2,4,6,8)])

### DerSimonian and Laird method using the treatment differences
res <- rma(yi, vi, data=dat2, method="DL", digits=2)
res

### Begg & Pilote (1991) model incorporating the uncontrolled studies
res <- rma.mv(yi, vi, mods = ~ trt, random = ~ 1 | study,
              data=dat, method="ML", digits=2)
res

### model involving bias terms for the uncontrolled studies
res <- rma.mv(yi, vi, mods = ~ trt + trt:arms, random = ~ 1 | study,
              data=dat, method="ML", digits=2)
res

### model with random treatment effect
res <- rma.mv(yi, vi, mods = ~ trt, random = list(~ 1 | study, ~ trt | study),
              struct="UN", tau2=c(0,NA), rho=0, data=dat, method="ML", digits=2)
res

### model with random treatment effect, but with equal variances in both arms
res <- rma.mv(yi, vi, mods = ~ trt, random = list(~ 1 | study, ~ trt | study),
              struct="CS", rho=0, data=dat, method="ML", digits=2)
res
```

---

dat.berkey1998          *Studies on Treatments for Periodontal Disease*

---

## Description

Results from 5 trials comparing surgical and non-surgical treatments for medium-severity periodontal disease one year after treatment.

## Usage

```
dat.berkey1998
```

**Format**

The data frame contains the following columns:

| **trial** | numeric | trial number |
| **author** | character | study author(s) |
| **year** | numeric | publication year |
| **ni** | numeric | number of patients |
| **outcome** | character | outcome (PD = probing depth; AL = attachment level) |
| **yi** | numeric | observed mean difference in outcome (surgical versus non-surgical) |
| **v1i** | numeric | variances and covariances of the observed effects |
| **v2i** | numeric | variances and covariances of the observed effects |

## Details

The dataset includes the results from 5 trials that compared surgical and non-surgical methods for the treatment of medium-severity periodontal disease. Reported outcomes include the change in probing depth (PD) and attachment level (AL) one year after the treatment. The effect size measure used for this meta-analysis was the (raw) mean difference, calculated in such a way that positive values indicate that surgery was more effective than non-surgical treatment in decreasing the probing depth and increasing the attachment level (so, the results from the various trials indicate that surgery is preferable for reducing the probing depth, while non-surgical treatment is preferable for increasing the attachment level). Since each trial provides effect size estimates for both outcomes, the estimates are correlated. A multivariate model can be used to meta-analyze the two outcomes simultaneously.

The v1i and v2i values are the variances and covariances of the observed effects. In particular, for each study, variables v1i and v2i form a $2 \times 2$ variance-covariance matrix of the observed effects, with the diagonal elements corresponding to the sampling variances of the mean differences (the first for probing depth, the second for attachment level) and the off-diagonal value corresponding to the covariance of the two mean differences. Below, the full (block diagonal) variance-covariance for all studies is constructed from these two variables.

## Source

Berkey, C. S., Antczak-Bouckoms, A., Hoaglin, D. C., Mosteller, F., & Pihlstrom, B. L. (1995). Multiple-outcomes meta-analysis of treatments for periodontal disease. *Journal of Dental Research*, **74**, 1030–1039.

Berkey, C. S., Hoaglin, D. C., Antczak-Bouckoms, A., Mosteller, F., & Colditz, G. A. (1998). Meta-analysis of multiple outcomes by regression with random effects. *Statistics in Medicine*, **17**, 2537–2550.

## Examples

```
### load data
dat <- get(data(dat.berkey1998))

### construct list of the variance-covariance matrices of the observed outcomes for the studies
V <- lapply(split(dat[,c("v1i", "v2i")], dat$trial), as.matrix)

### construct block diagonal matrix
V <- bldiag(V)

### fit multiple outcomes (meta-regression) model (with ML estimation)
```

```
res <- rma.mv(yi, V, mods = ~ outcome + outcome:I(year - 1983) - 1,
                random = ~ outcome | trial, struct="UN", data=dat, method="ML")
print(res, digits=3)
```

---

dat.bonett2010           *Studies on the Reliability of the CES-D Scale*

---

### Description

Results from 9 studies on the reliability of the Center for Epidemiologic Studies Depression (CES-D) Scale administered to children providing care to an elderly parent.

### Usage

```
dat.bonett2010
```

### Format

The data frame contains the following columns:

|  |  |  |
|---|---|---|
| **study** | numeric | study number |
| **source** | character | source of data |
| **ni** | numeric | sample size |
| **mi** | numeric | number of items in the scale |
| **ai** | numeric | observed value of Cronbach's alpha |
| **caregivers** | character | gender of the children in the sample |

### Details

The Center for Epidemiologic Studies Depression (CES-D) Scale is a 20-item questionnaire assessing various symptoms of depression, with each item scored on a 4-point scale. The scale has been used in several studies to examine depressive symptoms in children providing care to an elderly parent. The dataset includes information on the reliability of the scale as measured with Cronbach's alpha in 9 such studies. Also, the gender composition of the children in each sample is indicated.

### Source

Bonett, D. G. (2010). Varying coefficient meta-analytic methods for alpha reliability. *Psychological Methods*, **15**, 368–385.

### References

Bonett, D. G. (2002). Sample size requirements for testing and estimating coefficient alpha. *Journal of Educational and Behavioral Statistics*, **27**, 335–340.

Hakstian, A. R., & Whalen, T. E. (1976). A k-sample significance test for independent alpha coefficients. *Psychometrika*, **41**, 219–231.

## Examples

```
### load data
dat <- get(data(dat.bonett2010))

### meta-analysis using the raw alpha values
res <- rma(measure="ARAW", ai=ai, mi=mi, ni=ni, data=dat)
res

### meta-analysis using transformed alpha values (using the
### transformation suggested by Hakstian & Whalen, 1976)
res <- rma(measure="AHW", ai=ai, mi=mi, ni=ni, data=dat)
res
predict(res, transf=transf.iahw)

### meta-analysis using transformed alpha values (using the
### transformation suggested by Bonett, 2002)
res <- rma(measure="ABT", ai=ai, mi=mi, ni=ni, data=dat)
res
predict(res, transf=transf.iabt)

### examine whether female/mixed samples yield different alphas (with raw alphas)
res <- rma(measure="ARAW", ai=ai, mi=mi, ni=ni, mods = ~ caregivers, data=dat)
res
predict(res, newmods=c(0,1), digits=2)
```

---

dat.bourassa1996 *Studies on the Relationship between Handedness and Eye-Dominance*

---

## Description

Results from 47 studies on the relationship between handedness and eye-dominance.

## Usage

```
dat.bourassa1996
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **sample** | numeric | sample number |
| **author** | character | (first) author |
| **year** | numeric | publication year |
| **selection** | character | selection of subjects on the basis of eyedness or handedness |
| **investigator** | character | investigator (psychologist, educationalist, or other) |
| **hand_assess** | character | method to assess handedness (questionnaire or performance based) |
| **eye_assess** | character | method to assess eyedness (see below for classification) |
| **mage** | numeric | mean age of sample |

| **lh.le** | numeric   | number of left-handed left-eyed individuals       |
| **lh.re** | numeric   | number of left-handed right-eyed individuals      |
| **rh.le** | numeric   | number of right-handed left-eyed individuals      |
| **rh.re** | numeric   | number of right-handed right-eyed individuals     |
| **sex**   | character | sex of the sample (combined, male, or female)     |

### Details

The 47 studies included in this meta-analysis examined the relationship between handedness and eye-dominance (ocular dominance or eyedness). Results are given in terms of $2 \times 2$ tables, indicating the number of left-handed left-eyed, left-handed right-eyed, right-handed left-eyed, and right-handed right-eyed individuals. Note that some studies included multiple (independent) samples, so that the meta-analysis included 54 samples in total. Also, for some studies, the combined data of the males and females are further broken down into the two subgroups.

In some studies, there was indication that the selection of subjects was not random with respect to handedness and/or eyedness. While this should not influence the size of the association as measured with the odds ratio, this invalidates those studies for assessing the overall percentage of left-eyed and left-handed individuals.

Handedness was assessed in the individual studies either based on a questionnaire or inventory or based on task performance. Eyedness was assessed based on various methods: E.1 methods are based on task performance, while E.2.a denotes assessment based on a questionnaire. The performance based methods could be further broken down into: E.1.a.i (monocular procedure with object/instrument held in one hand), E.1.a.i (monocular procedure with object/instrument held in both hands), E.1.b (binocular procedure), E.1.c (a combination of the previous methods), and E.1.d (some other method).

### Source

Bourassa, D. C., McManus, I. C., & Bryden, M. P. (1996). Handedness and eye-dominance: A meta-analysis of their relationship. *Laterality*, **1**, 5–34.

### Examples

```
### load data
dat <- get(data(dat.bourassa1996))

### calculate log(OR) and corresponding sampling variance with 1/2 correction
dat <- escalc(measure="OR", ai=ai, bi=bi, ci=ci, di=di, data=dat, add=1/2, to="all")

### overall association between handedness and eyedness
res <- rma(yi, vi, data=dat, subset=sex=="combined")
res
predict(res, transf=exp, digits=2)
```

| dat.colditz1994 | *Studies on the Effectiveness of the BCG Vaccine Against Tuberculosis* |
|---|---|

### Description

Results from 13 studies examining the effectiveness of the Bacillus Calmette-Guerin (BCG) vaccine against tuberculosis.

### Usage

```
dat.colditz1994
dat.bcg
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **trial** | numeric | trial number |
| **author** | character | author(s) |
| **year** | numeric | publication year |
| **tpos** | numeric | number of TB positive cases in the treated (vaccinated) group |
| **tneg** | numeric | number of TB negative cases in the treated (vaccinated) group |
| **cpos** | numeric | number of TB positive cases in the control (non-vaccinated) group |
| **cneg** | numeric | number of TB negative cases in the control (non-vaccinated) group |
| **ablat** | numeric | absolute latitude of the study location (in degrees) |
| **alloc** | character | method of treatment allocation (random, alternate, or systematic assignment) |

### Details

The 13 studies provide data in terms of $2 \times 2$ tables in the form:

| | TB positive | TB negative |
|---|---|---|
| vaccinated group | tpos | tneg |
| control group | cpos | cneg |

The goal of the meta-analysis was to examine the overall effectiveness of the BCG vaccine for preventing tuberculosis and to examine moderators that may potentially influence the size of the effect.

The dataset has been used in several publications to illustrate meta-analytic methods (see 'References').

### Source

Colditz, G. A., Brewer, T. F., Berkey, C. S., Wilson, M. E., Burdick, E., Fineberg, H. V., & Mosteller, F. (1994). Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association*, **271**, 698–702.

**References**

Berkey, C. S., Hoaglin, D. C., Mosteller, F., & Colditz, G. A. (1995). A random-effects regression model for meta-analysis. *Statistics in Medicine*, **14**, 395–411.

van Houwelingen, H. C., Arends, L. R., & Stijnen, T. (2002). Advanced methods in meta-analysis: Multivariate approach and meta-regression. *Statistics in Medicine*, **21**, 589–624.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**Examples**

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
dat

### random-effects model
res <- rma(yi, vi, data=dat)
res

### average relative risk with 95% CI
predict(res, transf=exp)

### mixed-effects model with absolute latitude and publication year as moderators
res <- rma(yi, vi, mods = ~ ablat + year, data=dat)
res

### predicted average relative risks for 10-60 degrees absolute latitude
### holding the publication year constant at 1970
predict(res, newmods=cbind(seq(from=10, to=60, by=10), 1970), transf=exp)

### note: the interpretation of the results is difficult because absolute
### latitude and publication year are strongly correlated (the more recent
### studies were conducted closer to the equator)
plot(dat$ablat, dat$year)
cor(dat$ablat, dat$year)
```

---

dat.collins1985a          *Studies on the Treatment of Upper Gastrointestinal Bleeding by a His-*
                          *tamine H2 Antagonist*

---

**Description**

Results from studies examining the effectiveness of histamine H2 antagonists (cimetidine or raniti-dine) in treating patients with acute upper gastrointestinal hemorrhage.

## Usage

```
dat.collins1985a
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **id** | numeric | study number |
| **trial** | character | first author of trial |
| **year** | numeric | year of publication |
| **ref** | numeric | reference number |
| **trt** | character | C = cimetidine, R = ranitidine |
| **ctrl** | character | P = placebo, AA = antacids, UT = usual treatment |
| **nti** | numeric | number of patients in treatment group |
| **b.xti** | numeric | number of patients in treatment group with persistent or recurrent bleedings |
| **o.xti** | numeric | number of patients in treatment group in need of operation |
| **d.xti** | numeric | number of patients in treatment group that died |
| **nci** | numeric | number of patients in control group |
| **b.xci** | numeric | number of patients in control group with persistent or recurrent bleedings |
| **o.xci** | numeric | number of patients in control group in need of operation |
| **d.xci** | numeric | number of patients in control group that died |

## Details

The data were obtained from Tables 1 and 2 in Collins and Langman (1985). The authors used Peto's (one-step) method for meta-analyzing the 27 trials. This approach is implemented in the `rma.peto` function. Using the same dataset, van Houwelingen, Zwinderman, and Stijnen (1993) describe some alternative approaches for analyzing these data, including fixed and random-effects conditional logistic models. Those are implemented in the `rma.glmm` function.

## Source

Collins, R., & Langman, M. (1985). Treatment with histamine H2 antagonists in acute upper gastrointestinal hemorrhage. *New England Journal of Medicine*, **313**, 660–666.

## References

van Houwelingen, H. C., Zwinderman, K. H., & Stijnen, T. (1993). A bivariate approach to meta-analysis. *Statistics in Medicine*, **12**, 2273–2284.

## Examples

```
### load data
dat <- get(data(dat.collins1985a))

### meta-analysis of log ORs using Peto's method (outcome: persistent or recurrent bleedings)
res <- rma.peto(ai=b.xti, n1i=nti, ci=b.xci, n2i=nci, data=dat)
print(res, digits=2)

## Not run:
```

```
### meta-analysis of log ORs using a conditional logistic regression model (FE model)
res <- rma.glmm(measure="OR", ai=b.xti, n1i=nti, ci=b.xci, n2i=nci, data=dat,
                model="CM.EL", method="FE")
summary(res)
predict(res, transf=exp, digits=2)

### plot the log-likelihoods of the odds ratios
llplot(measure="OR", ai=b.xti, n1i=nti, ci=b.xci, n2i=nci, data=dat,
       lwd=1, refline=NA, xlim=c(-4,4), drop00=FALSE)

### meta-analysis of log odds ratios using a conditional logistic regression model (RE model)
res <- rma.glmm(measure="OR", ai=b.xti, n1i=nti, ci=b.xci, n2i=nci, data=dat,
                model="CM.EL", method="ML")
summary(res)
predict(res, transf=exp, digits=2)

## End(Not run)

### meta-analysis of log ORs using Peto's method (outcome: need for surgery)
res <- rma.peto(ai=o.xti, n1i=nti, ci=o.xci, n2i=nci, data=dat)
print(res, digits=2)

### meta-analysis of log ORs using Peto's method (outcome: death)
res <- rma.peto(ai=d.xti, n1i=nti, ci=d.xci, n2i=nci, data=dat)
print(res, digits=2)
```

---

dat.collins1985b    *Studies on the Effects of Diuretics in Pregnancy*

---

### Description

Results from 9 studies examining the effects of diuretics in pregnancy on various outcomes.

### Usage

    dat.collins1985b

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **id** | numeric | study number |
| **author** | character | study author(s) |
| **year** | numeric | publication year |
| **pre.nti** | numeric | number of women in treatment group followed up for pre-eclampsia outcome |
| **pre.nci** | numeric | number of women in control/placebo group followed up for pre-eclampsia outcome |
| **pre.xti** | numeric | number of women in treatment group with any form of pre-eclampsia |
| **pre.xci** | numeric | number of women in control/placebo group with any form of pre-eclampsia |
| **oedema** | numeric | dummy variable indicating whether oedema was a diagnostic criterion |

| **fup.nti** | numeric | number of women in treatment group followed up for mortality outcomes |
| **fup.nci** | numeric | number of women in control/placebo group followed up for mortality outcomes |
| **ped.xti** | numeric | number of perinatal deaths in treatment group |
| **ped.xci** | numeric | number of perinatal deaths in control/placebo group |
| **stb.xti** | numeric | number of stillbirths in treatment group |
| **stb.xci** | numeric | number of stillbirths in control/placebo group |
| **ned.xti** | numeric | number of neonatal deaths in treatment group |
| **ned.xci** | numeric | number of neonatal deaths in control/placebo group |

## Details

The 9 studies in this dataset examined the effects of diuretics in pregnancy on various outcomes, including the presence of any form of pre-eclampsia, perinatal death, stillbirth, and neonatal death.

## Source

Collins, R., Yusuf, S., & Peto, R. (1985). Overview of randomised trials of diuretics in pregnancy. *British Medical Journal*, **290**, 17–23.

## Examples

```
### load data
dat <- get(data(dat.collins1985b))

### calculate (log) odds ratio and sampling variance
dat <- escalc(measure="OR", n1i=pre.nti, n2i=pre.nci, ai=pre.xti, ci=pre.xci, data=dat)
summary(dat, digits=2, transf=exp)

### meta-analysis using Peto's method for any form of pre-eclampsia
rma.peto(n1i=pre.nti, n2i=pre.nci, ai=pre.xti, ci=pre.xci, data=dat, digits=2)

### meta-analysis including only studies where oedema was not a diagnostic criterion
rma.peto(n1i=pre.nti, n2i=pre.nci, ai=pre.xti, ci=pre.xci, data=dat, digits=2, subset=(oedema==0))

### meta-analyses of mortality outcomes (perinatal deaths, stillbirths, and neonatal deaths)
rma.peto(n1i=fup.nti, n2i=fup.nci, ai=ped.xti, ci=ped.xci, data=dat, digits=2)
rma.peto(n1i=fup.nti, n2i=fup.nci, ai=stb.xti, ci=stb.xci, data=dat, digits=2)
rma.peto(n1i=fup.nti, n2i=fup.nci, ai=ned.xti, ci=ned.xci, data=dat, digits=2)
```

---

dat.curtis1998        *Studies on the Effects of Elevated CO2 Levels on Woody Plant Mass*

---

## Description

Results from studies examining the effects of elevated CO2 levels on woody plant mass.

## Usage

```
dat.curtis1998
```

**Format**

The data frame contains the following columns:

| | | |
|---|---|---|
| **id** | numeric | observation number |
| **paper** | numeric | paper number |
| **units** | numeric | paper number |
| **genus** | character | genus name |
| **species** | character | species name |
| **function** | character | plant functional group |
| **co2.ambi** | numeric | ambient CO2 level (control group) |
| **co2.elev** | numeric | elevated CO2 level (treatment group) |
| **units** | character | units for CO2 exposure levels |
| **time** | numeric | maximum length of time (days) of CO2 exposure |
| **pot** | character | growing method (see below) |
| **method** | character | CO2 exposure facility (see below) |
| **stock** | character | planting stock code |
| **xtrt** | character | interacting treatment code (see below) |
| **level** | character | interacting treatment level codes (see below) |
| **m1i** | numeric | mean plant mass under elevated CO2 level (treatment group) |
| **sd1i** | numeric | standard deviation of plant mass underelevated CO2 level (treatment group) |
| **n1i** | numeric | number of observations under elevated CO2 level (treatment group) |
| **m2i** | numeric | mean plant mass under ambient CO2 level (control group) |
| **sd2i** | numeric | standard deviation of plant mass under ambient CO2 level (control group) |
| **n2i** | numeric | number of observations under ambient CO2 level (control group) |

**Details**

The studies included in this dataset compared the total above- plus below-ground biomass (in grams) for plants that were either exposed to ambient (around 35 Pa) and elevated CO2 levels (around twice the ambient level). The co2.ambi and co2.elev variables indicate the CO2 levels in the control and treatment groups, respectively (with the units variable specifying the units for the CO2 exposure levels). Many of the studies also varied one or more additional environmental variables (defined by the xtrt and level variables):

- NONE = no additional treatment factor
- FERT = soil fertility (either a CONTROL, HIGH, or LOW level)
- LIGHT = light treatment (always a LOW light level)
- FERT+L = soil fertility and light (a LOW light and soil fertility level)
- H2O = well watered vs drought (either a WW or DRT level)
- TEMP = temperature treatment (either a HIGH or LOW level)
- OZONE = ozone exposure (either a HIGH or LOW level)
- UVB = ultraviolet-B radiation exposure (either a HIGH or LOW level)

In addition, the studies differed with respect to various design variables, including CO2 exposure duration (time), growing method (pot: number = pot size in liters; GRND = plants rooted in ground; HYDRO = solution or aeroponic culture), CO2 exposure facility (method: GC = growth chamber; GH = greenhouse; OTC = field-based open-top chamber), and planting stock (stock: SEED = plants started

from seeds; SAP = plants started from cuttings). The goal of the meta-analysis was to examine the effects of elevated CO2 levels on plant physiology and growth and the interacting effects of the environmental (and design) variables.

## Source

Hedges, L. V., Gurevitch, J., & Curtis, P. S. (1999). The meta-analysis of response ratios in experimental ecology. *Ecology*, **80**, 1150–1156. (data obtained from *Ecological Archives*, E080-008-S1, at: <http://www.esapubs.org/archive/ecol/E080/008/>)

## References

Curtis, P. S., & Wang, X. (1998). A meta-analysis of elevated CO2 effects on woody plant mass, form, and physiology. *Oecologia*, **113**, 299–313.

## Examples

```
### load data
dat <- get(data(dat.curtis1998))

### calculate log ratio of means and corresponding sampling variances
dat <- escalc(measure="ROM", m1i=m1i, sd1i=sd1i, n1i=n1i, m2i=m2i, sd2i=sd2i, n2i=n2i, data=dat)
dat

### meta-analysis of log ratio of means using a random-effects model
res <- rma(yi, vi, method="DL", data=dat)
res

### average ratio of means with 95% CI
predict(res, transf=exp, digits=2)

### meta-analysis for plants grown under nutrient stress
res <- rma(yi, vi, method="DL", data=dat, subset=(xtrt=="FERT" & level=="LOW"))
predict(res, transf=exp, digits=2)

### meta-analysis for plants grown under low light conditions
res <- rma(yi, vi, method="DL", data=dat, subset=(xtrt=="LIGHT" & level=="LOW"))
predict(res, transf=exp, digits=2)
```

---

dat.debruin2009 *Studies on Standard Care Quality and HAART-Adherence*

---

## Description

Results from 13 trials providing information about standard care quality and HAART-adherence in control groups.

## Usage

```
dat.debruin2009
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **author** | character | (first) author of study |
| **year** | numeric | publication year |
| **scq** | numeric | standard care quality |
| **ni** | numeric | number of patients in standard care group |
| **xi** | numeric | number of patients with undetectable viral load in standard care group |
| **mi** | numeric | number of patients with detectable viral load in standard care group |
| **ethnicity** | numeric | dominant ethnicity of the patients in standard care group |
| **patients** | numeric | inclusion of patients continuing or starting (a new) treatment |
| **select** | numeric | baseline selection of patients with adherence problems or no selection |
| **sens** | numeric | sensitivity of viral load assessments (<400 vs. >=400 copies/ml). |

## Details

Highly active antiretroviral therapy (HAART) refers to a combination of multiple antiretroviral drugs that can effectively suppress the HIV virus. However, achieving viral suppression (to the point that the virus becomes essentially undetectable in a blood sample) requires high levels of adherence to an often complicated medication regimen. A number of trials have examined various interventions that aim to increase adherence levels. In each trial, patients receiving the intervention are compared to patients in a control group receiving standard care (often referred to as 'care as usual'). However, the quality of standard care can vary substantially between these studies. de Bruin et al. (2009) assessed the quality of standard care provided (based on a quantification of the number of behavior change techniques applied) and examined to what extent the quality of standard care was related to the proportion of patients achieving effective viral suppression in the control groups.

## Source

de Bruin, M., Viechtbauer, W., Hospers, H. J., Schaalma, H. P., & Kok, G. (2009). Standard care quality determines treatment outcomes in control groups of HAART-adherence intervention studies: Implications for the interpretation and comparison of intervention effects. *Health Psychology*, **28**, 668–674.

## Examples

```
### load data
dat <- get(data(dat.debruin2009))

### calculate proportions and corresponding sampling variances
dat <- escalc(measure="PR", xi=xi, ni=ni, data=dat)

### random-effects model
res <- rma(yi, vi, data=dat)
print(res, digits=2)

### mixed-effects meta-regression model with all predictors/covariates
res <- rma(yi, vi, mods = ~ scq + ethnicity + patients + select + sens, data=dat)
print(res, digits=3)
```

```
### mixed-effects meta-regression model with scq and ethnicity as predictors/covariates
res <- rma(yi, vi, mods = ~ scq + ethnicity, data=dat)
print(res, digits=3)
```

---

| dat.fine1993 | *Studies on Radiation Therapy with or without Adjuvant Chemotherapy in Patients with Malignant Gliomas* |
|---|---|

---

## Description

Results from 17 trials comparing post-operative radiation therapy with and without adjuvant chemotherapy in patients with malignant gliomas.

## Usage

```
dat.fine1993
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **nei** | numeric | sample size in the experimental group receiving radiotherapy plus adjuvant chemotherapy |
| **nci** | numeric | sample size in the control group receiving radiotherapy alone |
| **e1i** | numeric | number of survivors at 6 months in the experimental group |
| **c1i** | numeric | number of survivors at 6 months in the control group |
| **e2i** | numeric | number of survivors at 12 months in the experimental group |
| **c2i** | numeric | number of survivors at 12 months in the control group |
| **e3i** | numeric | number of survivors at 18 months in the experimental group |
| **c3i** | numeric | number of survivors at 18 months in the control group |
| **e4i** | numeric | number of survivors at 24 months in the experimental group |
| **c4i** | numeric | number of survivors at 24 months in the control group |

## Details

The 17 trials report the post-operative survival of patients with malignant gliomas receiving either radiation therapy with adjuvant chemotherapy or radiation therapy alone. Survival was assessed at 6, 12, 18, and 24 months in all but one study (which assessed survival only at 12 and at 24 months).

The data were reconstructed by Trikalinos and Olkin (2012) based on Table 2 in Fine et al. (1993) and Table 3 in Dear (1994). The data can be used to illustrate how a meta-analysis can be conducted of effect sizes reported at multiple time points using a multivariate model.

## Source

Dear, K. B. G. (1994). Iterative generalized least squares for meta-analysis of survival data at multiple times. *Biometrics*, **50**, 989–1002.

Trikalinos, T. A., & Olkin, I. (2012). Meta-analysis of effect sizes reported at multiple time points: A multivariate approach. *Clinical Trials*, **9**, 610–620.

### References

Fine, H. A., Dear, K. B., Loeffler, J. S., Black, P. M., & Canellos, G. P. (1993). Meta-analysis of radiation therapy with and without adjuvant chemotherapy for malignant gliomas in adults. *Cancer*, **71**, 2585–2597.

### Examples

```
### load data
dat <- get(data(dat.fine1993))

### calculate log(ORs) and sampling variances for each time point
dat <- escalc(measure="OR", ai=e1i, n1i=nei, ci=c1i, n2i=nci, data=dat, var.names=c("y1i","v1i"))
dat <- escalc(measure="OR", ai=e2i, n1i=nei, ci=c2i, n2i=nci, data=dat, var.names=c("y2i","v2i"))
dat <- escalc(measure="OR", ai=e3i, n1i=nei, ci=c3i, n2i=nci, data=dat, var.names=c("y3i","v3i"))
dat <- escalc(measure="OR", ai=e4i, n1i=nei, ci=c4i, n2i=nci, data=dat, var.names=c("y4i","v4i"))

### calculate the covariances (equations in Appendix of Trikalinos & Olkin, 2012)
dat$v12i <- with(dat, nei / (e1i * (nei - e2i)) + nci / (c1i * (nci - c2i)))
dat$v13i <- with(dat, nei / (e1i * (nei - e3i)) + nci / (c1i * (nci - c3i)))
dat$v14i <- with(dat, nei / (e1i * (nei - e4i)) + nci / (c1i * (nci - c4i)))
dat$v23i <- with(dat, nei / (e2i * (nei - e3i)) + nci / (c2i * (nci - c3i)))
dat$v24i <- with(dat, nei / (e2i * (nei - e4i)) + nci / (c2i * (nci - c4i)))
dat$v34i <- with(dat, nei / (e3i * (nei - e4i)) + nci / (c3i * (nci - c4i)))

### create dataset in long format
dat.long <- data.frame(study=rep(1:nrow(dat), each=4), time=1:4,
                       yi=c(t(dat[,c("y1i","y2i","y3i","y4i")])),
                       vi=c(t(dat[,c("v1i","v2i","v3i","v4i")])))

### var-cov matrices of the sudies
V <- lapply(split(dat, dat$study),
            function(x) matrix(c( x$v1i, x$v12i, x$v13i, x$v14i,
                                  x$v12i,  x$v2i, x$v23i, x$v24i,
                                  x$v13i, x$v23i,  x$v3i, x$v34i,
                               x$v14i, x$v24i, x$v34i,  x$v4i), nrow=4, ncol=4, byrow=TRUE))

### remove rows for the missing time points in study 17
dat.long <- na.omit(dat.long)

### remove corresponding rows/columns from var-cov matrix
V[[17]] <- V[[17]][c(2,4),c(2,4)]

### make a copy of V
Vc <- V

### replace any (near) singular var-cov matrices with ridge corrected versions
repl.Vi <- function(Vi) {
   res <- eigen(Vi)
```

```
      if (any(res$values <= .08)) {
         round(res$vectors %*% diag(res$values + .08) %*% t(res$vectors), 12)
      } else {
         Vi
      }
   }
}
Vc <- lapply(Vc, repl.Vi)

### do not correct var-cov matrix of study 17
Vc[[17]] <- V[[17]]

### construct block diagonal matrix
Vc <- bldiag(Vc)

### multivariate fixed-effects model
res <- rma.mv(yi, Vc, mods = ~ factor(time) - 1, method="FE", data=dat.long)
print(res, digits=3)

### multivariate random-effects model with heteroscedastic AR(1) structure for the true effects
res <- rma.mv(yi, Vc, mods = ~ factor(time) - 1, random = ~ time | study,
              struct="HAR", data=dat.long)
print(res, digits=3)

## Not run:
### profile the variance components
par(mfrow=c(2,2))
profile(res, tau2=1, xlim=c(0,.2))
profile(res, tau2=2, xlim=c(0,.2))
profile(res, tau2=3, xlim=c(0,.2))
profile(res, tau2=4, xlim=c(.1,.3))

## End(Not run)
## Not run:
### profile the autocorrelation coefficient
par(mfrow=c(1,1))
profile(res, rho=1)

## End(Not run)
```

---

| dat.gibson2002 | *Studies on the Effectiveness of Self-Management Education and Regular Medical Review for Adults with Asthma* |
| --- | --- |

---

### Description

Results from 15 trials examining the effectiveness of self-management education and regular medical review for adults with asthma.

### Usage

```
dat.gibson2002
```

**Format**

The data frame contains the following columns:

| **author** | character | first author of study |
|---|---|---|
| **year** | numeric | publication year |
| **n1i** | numeric | number of participants in the intervention group |
| **m1i** | numeric | mean number of days off work/school in the intervention group |
| **sd1i** | numeric | standard deviation of the number of days off work/school in the intervention group |
| **n2i** | numeric | number of participants in the control/comparison group |
| **m2i** | numeric | mean number of days off work/school in the control/comparison group |
| **sd2i** | numeric | standard deviation of the number of days off work/school in the control/comparison group |
| **ai** | numeric | number of participants who had one or more days off work/school in the intervention group |
| **bi** | numeric | number of participants who no days off work/school in the intervention group |
| **ci** | numeric | number of participants who had one or more days off work/school in the control/comparison group |
| **di** | numeric | number of participants who no days off work/school in the control/comparison group |
| **type** | numeric | numeric code for the intervention type (see below) |

**Details**

Asthma management guidelines typically recommend for patients to receive education and regular medical review. While self-management programs have been shown to increase patient knowledge, it is less clear to what extent they actually impact health outcomes. The systematic review by Gibson et al. (2002) examined the effectiveness of self-management education and regular medical review for adults with asthma. In each study, participants receiving a certain management intervention were compared against those in a control/comparison group with respect to a variety of health outcomes. One of the outcomes examined in a number of studies was the number of days off work/school.

The majority of studies reporting this outcome provided means and standard deviations allowing a meta-analysis of standardized mean differences. Seven studies also reported the number of participants who had one or more days off work/school in each group. These studies could be meta-analyzed using, for example, (log) risk ratios. Finally, one could also consider a combined analysis based on standardized mean differences computed from the means and standard deviations where available and using probit transformed risk differences (which also provide estimates of the standardized mean difference) for the remaining studies.

Some degree of patient education was provided in all studies. In addition, the type variable indicates what additional intervention components were included in each study:

1. optimal self-management (writing action plan, self-monitoring, regular medical review),
2. self-monitoring and regular medical review,
3. self-monitoring only,
4. regular medical review only,
5. written action plan only.

**Source**

Gibson, P. G., Powell, H., Wilson, A., Abramson, M. J., Haywood, P., Bauman, A., et al. (2002). Self-management education and regular practitioner review for adults with asthma. *Cochrane Database of Systematic Reviews*, **3**.

## Examples

```
### load data
dat <- get(data(dat.gibson2002))

### fixed-effects model analysis of the standardized mean differences
dat <- escalc(measure="SMD", m1i=m1i, sd1i=sd1i, n1i=n1i, m2i=m2i, sd2i=sd2i, n2i=n2i, data=dat)
res <- rma(yi, vi, data=dat, method="FE")
print(res, digits=2)

### fixed-effects model analysis of the (log) risk ratios
dat <- escalc(measure="RR", ai=ai, bi=bi, ci=ci, di=di, data=dat)
res <- rma(yi, vi, data=dat, method="FE")
print(res, digits=2)
predict(res, transf=exp, digits=2)

### fixed-effects model analysis of the standardized mean differences and the probit transformed
### risk differences (which also provide estimates of the standardized mean difference)
dat <- escalc(measure="SMD", m1i=m1i, sd1i=sd1i, n1i=n1i, m2i=m2i, sd2i=sd2i, n2i=n2i, data=dat)
dat <- escalc(measure="PBIT", ai=ai, bi=bi, ci=ci, di=di, data=dat, replace=FALSE)
dat
res <- rma(yi, vi, data=dat, method="FE")
print(res, digits=2)
```

---

dat.hackshaw1998          *Studies on Lung Cancer Risk from ETS Exposure*

---

## Description

Results from 37 studies on the risk of lung cancer from environmental tobacco smoke (ETS) exposure.

## Usage

```
dat.hackshaw1998
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **author** | character | first author of study |
| **year** | numeric | publication year |
| **country** | character | country where study was conducted |
| **design** | character | either cohort or case-control study |
| **cases** | numeric | number of lung cancer cases |
| **yi** | numeric | log odds ratio |
| **vi** | numeric | corresponding sampling variance |

## Details

The dataset includes the results from 37 studies (4 cohort, 33 case-control) on the risk of lung cancer from environmental tobacco smoke (ETS) exposure from the spouse in women who are lifelong nonsmokers. Values of the log odds ratio greater than 0 indicate an increased risk of cancer in exposed women compared to women not exposed to ETS from their spouse.

## Source

Hackshaw, A. K., Law, M. R., & Wald, N. J. (1997). The accumulated evidence on lung cancer and environmental tobacco smoke. *British Medical Journal*, **315**, 980–988.

Hackshaw, A. K. (1998). Lung cancer and passive smoking. *Statistical Methods in Medical Research*, **7**, 119–136.

## Examples

```
### load data
dat <- get(data(dat.hackshaw1998))

### random-effects model using the log odds ratios
res <- rma(yi, vi, data=dat, method="DL")
res

### estimated average odds ratio with CI (and credibility/prediction interval)
predict(res, transf=exp, digits=2)
```

---

dat.hart1999                          *Studies on the Effectiveness of Warfarin for Preventing Strokes*

---

## Description

Results from 6 clinical trials examining the effectiveness of adjusted-dose warfarin for preventing strokes in patients with atrial fibrillation.

## Usage

```
dat.hart1999
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **trial** | numeric | trial number |
| **study** | character | study name (abbreviated) |
| **year** | numeric | publication year |
| **x1i** | numeric | number of strokes in the warfarin group |
| **n1i** | numeric | number of patients in the warfarin group |
| **t1i** | numeric | total person-time (in years) in the warfarin group |
| **x2i** | numeric | number of strokes in the placebo/control group |

| | | |
|---|---|---|
| **n2i** | numeric | number of patients in the placebo/control group |
| **t2i** | numeric | total person-time (in years) in the placebo/control group |
| **compgrp** | character | type of comparison group (placebo or control) |
| **prevtype** | character | type of prevention (primary or secondary) |
| **trinr** | character | target range for the international normalized ratio (INR) |

## Details

The 6 studies provide data with respect to the number of strokes in the warfarin and the comparison (placebo or control) group. In addition, the number of patients and the total person-time (in years) is provided for the two groups. The goal of the meta-analysis was to examine the effectiveness of adjusted-dose warfarin for preventing strokes in patients with atrial fibrillation.

## Source

Hart, R. G., Benavente, O., McBride, R., & Pearce, L. A. (1999). Antithrombotic therapy to prevent stroke in patients with atrial fibrillation: A meta-analysis. *Annals of Internal Medicine*, **131**, 492–501.

## Examples

```
### load data
dat <- get(data(dat.hart1999))

### calculate log incidence rate ratios and corresponding sampling variances
dat <- escalc(measure="IRR", x1i=x1i, x2i=x2i, t1i=t1i, t2i=t2i, data=dat)
dat

### meta-analysis of log incidence rate ratios using a random-effects model
res <- rma(yi, vi, data=dat)
res

### average incidence rate ratio with 95% CI
predict(res, transf=exp)

### forest plot with extra annotations
forest(res, xlim=c(-14, 6), at=log(c(.05, .25, 1, 4)), atransf=exp,
       slab=paste(dat$study, " (", dat$year, ")", sep=""),
       ilab=cbind(paste(dat$x1i, "/", dat$t1i, sep=" "),
       paste(dat$x2i, "/", dat$t2i, sep=" ")),
       ilab.xpos=c(-8,-5), cex=.85)
op <- par(cex=.85, font=2)
text(-14, 7.5, "Study (Year)", pos=4)
text(6,    7.5, "IRR [95% CI]", pos=2)
text(c(-8,-5), 8.0, c("Strokes /", "Strokes /"))
text(c(-8,-5), 7.5, c("Person-Time", "Person-Time"))
text(c(-8,-5), 8.5, c("Warfarin", "Control"))
segments(x0=-9, y0=8.25, x1=-4, y1=8.25)
par(op)

### meta-analysis of incidence rate differences using a random-effects model
res <- rma(measure="IRD", x1i=x1i, x2i=x2i, t1i=t1i, t2i=t2i, data=dat)
```

`res`

---

`dat.hasselblad1998`          *Studies on the Effectiveness of Counseling for Smoking Cessation*

---

### Description

Results from 24 studies on the effectiveness of various counseling types for smoking cessation.

### Usage

`dat.hasselblad1998`

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **id** | numeric | id number for each treatment arm |
| **study** | numeric | study id number |
| **authors** | character | study author(s) |
| **year** | numeric | publication year |
| **trt** | character | intervention group |
| **xi** | numeric | number of individuals abstinent |
| **ni** | numeric | number of individuals in group |

### Details

The dataset includes the results from 24 studies on the effectiveness of various counseling types for smoking cessation (i.e., self-help, individual counseling, group counseling, and no contact). The dataset indicates the total number of individuals within each study arm and the number that were abstinent from 6 to 12 months. The majority of the studies compared two interventions types against each other, while 2 studies compared three types against each other simultaneously.

The data can be used for a 'network meta-analysis' (also called 'mixed treatment comparison meta-analysis'). The code below shows how such an analysis can be conducted using an arm-based and a contrast-based model (see Salanti et al., 2008, for more details).

### Source

Hasselblad, V. (1998). Meta-analysis of multitreatment studies. *Medical Decision Making*, **18**, 37–43.

### References

Gleser, L. J., & Olkin, I. (2009). Stochastically dependent effect sizes. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 357–376). New York: Russell Sage Foundation.

Salanti, G., Higgins, J. P. T., Ades, A. E., & Ioannidis, J. P. A. (2008). Evaluation of networks of randomized trials. *Statistical Methods in Medical Research*, **17**, 279–301.

**Examples**

```
### load data
data(dat.hasselblad1998)

### calculate log odds for each study arm
dat <- escalc(measure="PLO", xi=xi, ni=ni, add=1/2, to="all", data=dat.hasselblad1998)

### create network graph (using 'plyr' and 'igraph' packages if installed)
if (require(plyr) && require(igraph)) {
   pairs <- do.call(rbind, sapply(split(dat$trt, dat$study), function(x) t(combn(x,2))))
   pairs <- ddply(data.frame(pairs), .(X1, X2), count)
   g <- graph.edgelist(as.matrix(pairs[,1:2]), directed=FALSE)
   plot(g, edge.curved=FALSE, edge.width=pairs$freq, vertex.label.dist=.7,
   vertex.label=c("Individual\nCounseling", "Group\nCounseling", "No Contact", "Self-Help"))
}

### convert trt variable to factor with desired ordering of levels
dat$trt <- factor(dat$trt, levels=c("no_contact", "self_help", "ind_counseling", "grp_counseling"))

### add a space before each level (this makes the output a bit more legible)
levels(dat$trt) <- paste0(" ", levels(dat$trt))

### network meta-analysis using an arm-based model with fixed study effects
### by setting rho=1/2, tau^2 reflects the amount of heterogeneity for all treatment comparisons
res <- rma.mv(yi, vi, mods = ~ factor(study) + trt - 1,
              random = ~ trt | study, rho=1/2, data=dat, btt=25:27)
res

### all pairwise odds ratios of interventions versus no contact
predict(res, newmods=cbind(matrix(0, nrow=3, ncol=24), diag(3)),
        intercept=FALSE, transf=exp, digits=2)

### all pairwise odds ratios comparing interventions (ic vs sh, gc vs sh, and gc vs ic)
predict(res, newmods=cbind(matrix(0,nrow=3,ncol=24), rbind(c(-1,1,0), c(-1,0,1), c(0,-1,1))),
        intercept=FALSE, transf=exp, digits=2)

### forest plot of ORs of interventions versus no contact
dev.new(width=7, height=4)
par(mar=c(5,4,1,1))
forest(c(0,res$b[25:27]), sei=c(0,res$se[25:27]), psize=1, xlim=c(-3,4), digits=c(2,1), efac=2,
       slab=c("No Contact", "Self-Help", "Individual Counseling", "Group Counseling"),
     atransf=exp, at=log(c(.5, 1, 2, 4, 8)), xlab="Odds Ratio for Intervention vs. No Contact")
text(-3, 6, "Intervention", pos=4)
text( 4, 6, "Odds Ratio [95% CI]",  pos=2)

### restructure dataset to a contrast-based format
dat.c <- lapply(split(dat.hasselblad1998, dat.hasselblad1998$study),
                function(x) cbind(x[rep(1,nrow(x)-1),], x[-1,5:7]))
dat.c <- do.call("rbind", dat.c)
```

```
dat.c <- dat.c[,c(1:5,8,6:7,9:10)]
names(dat.c)[5:10] <- c("trt1", "trt2", "ai", "n1i", "ci", "n2i")
rownames(dat.c) <- 1:nrow(dat.c)
dat.c$id <- 1:nrow(dat.c)
dat.c

### calculate log odds ratios for each treatment comparison
dat.c <- escalc(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, add=1/2, to="all", data=dat.c)
dat.c

### calculate the variance-covariance matrix of the log odds ratios for multitreatment studies
### see Gleser & Olkin (2009), equation (19.11), for the covariance equation
calc.v <- function(x) {
   v <- matrix(1/(x$ai[1]+1/2) + 1/(x$n1i[1] - x$ai[1] + 1/2), nrow=nrow(x), ncol=nrow(x))
   diag(v) <- x$vi
   v
}
V <- bldiag(lapply(split(dat.c, dat.c$study), calc.v))

### convert trt1 and trt2 variables to factors with desired ordering of levels
lvls <- c("no_contact", "self_help", "ind_counseling", "grp_counseling")
dat.c$trt1 <- factor(dat.c$trt1, levels=lvls)
dat.c$trt2 <- factor(dat.c$trt2, levels=lvls)

### create variables to indicate the contrast examined
dat.c <- cbind(dat.c, model.matrix(~ dat.c$trt1 - 1) - model.matrix(~ dat.c$trt2 - 1))
names(dat.c)[(ncol(dat.c)-3):ncol(dat.c)] <- lvls

### network meta-analysis using a contrast-based model
### by setting rho=1/2, tau^2 reflects the amount of heterogeneity for all treatment comparisons
res <- rma.mv(yi, V, mods = ~ self_help + ind_counseling + grp_counseling - 1,
              random = ~ factor(id) | study, rho=1/2, data=dat.c)
res
```

---

dat.hine1989                     *Studies on Prophylactic Use of Lidocaine After a Heart Attack*

---

### Description

Results from 6 studies evaluating mortality from prophylactic use of lidocaine in acute myocardial infarction.

### Usage

```
dat.hine1989
```

### Format

The data frame contains the following columns:

| **study** | numeric | study number |
| **source** | character | source of data |
| **n1i** | numeric | number of patients in lidocaine group |
| **n2i** | numeric | number of patients in control group |
| **ai** | numeric | number of deaths in lidocaine group |
| **ci** | numeric | number of deaths in control group |

## Details

Hine et al. (1989) conducted a meta-analysis of death rates in randomized controlled trials in which prophylactic lidocaine was administered to patients with confirmed or suspected acute myocardial infarction. The dataset describes the mortality at the end of the assigned treatment period for control and intravenous lidocaine treatment groups for six studies. The question of interest is whether there is a detrimental effect of lidocaine. Because the studies were conducted to compare rates of arrhythmias following a heart attack, the studies, taken individually, are too small to detect important differences in mortality rates.

The data in this dataset were obtained from Table I in Normand (1999, p. 322).

## Source

Normand, S. T. (1999). Meta-analysis: Formulating, evaluating, combining, and reporting. *Statistics in Medicine*, **18**, 321–359.

## References

Hine, L. K., Laird, N., Hewitt, P., & Chalmers, T. C. (1989). Meta-analytic evidence against prophylactic use of lidocaine in acute myocardial infarction. *Archives of Internal Medicine*, **149**, 2694–2698.

## Examples

```
### load data
dat <- get(data(dat.hine1989))

### calculate risk differences and corresponding sampling variances
dat <- escalc(measure="RD", n1i=n1i, n2i=n2i, ai=ai, ci=ci, data=dat)
dat

### meta-analysis of risk differences using a random-effects model
res <- rma(yi, vi, data=dat)
res
```

---

dat.ishak2007                     *Studies on Deep-Brain Stimulation*

---

## Description

Results from 46 studies examining the effects of deep-brain stimulation on motor skills of patients with Parkinson's disease.

## Usage

```
dat.ishak2007
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | character | (first) author and year |
| **y1i** | numeric | observed mean difference at 3 months |
| **v1i** | numeric | sampling variance of the mean difference at 3 months |
| **y2i** | numeric | observed mean difference at 6 months |
| **v2i** | numeric | sampling variance of the mean difference at 6 months |
| **y3i** | numeric | observed mean difference at 12 months |
| **v3i** | numeric | sampling variance of the mean difference at 12 months |
| **y4i** | numeric | observed mean difference at the long-term follow-up |
| **v4i** | numeric | sampling variance of the mean difference at the long-term follow-up |
| **mdur** | numeric | mean disease duration (in years) |
| **mbase** | numeric | mean baseline UPDRS score |

## Details

Deep-brain stimulation (DBS), which is delivered through thin surgically implanted wires in specific areas of the brain and controlled by the patient, is meant to provide relief of the debilitating symptoms of Parkinson's disease. The dataset includes the results from 46 studies examining the effects of DBS of the subthalamic nucleus on motor functioning, measured with the Unified Parkinson's Disease Rating Scale (UPDRS). The effect size measure for this meta-analysis was the mean difference of the scores while the stimulator is active and the baseline scores (before implantation of the stimulator). Since lower scores on the UPDRS indicate better functioning, negative numbers indicate improvements in motor skills. Effects were generally measured at 3, 6, and 12 months after implantation of the stimulator, with some studies also including a further long-term follow-up. However, the number of measurements differed between studies - hence the missing data on some of the measurement occasions.

Since the same patients were followed over time within a study, effect size estimates from multiple measurement occasions are likely to be correlated. A multivariate model accounting for the correlation in the effects can be used to meta-analyze these data. A difficulty with this approach is the lack of information about the correlation of the measurements over time in the individual studies. The approach taken by Ishak et al. (2007) was to assume an autoregressive (AR1) structure for the estimates within the individual studies. In addition, the correlation in the true effects was modeled, again using an autoregressive structure.

## Source

Ishak, K. J., Platt, R. W., Joseph, L., Hanley, J. A., & Caro, J. J. (2007). Meta-analysis of longitudinal studies. *Clinical Trials*, **4**, 525–539.

## Examples

```
### load data
dat <- get(data(dat.ishak2007))
```

```
### create long format dataset
dat.long <- reshape(dat, direction="long", idvar="study", v.names=c("yi","vi"),
                    varying=list(c(2,4,6,8), c(3,5,7,9)))
dat.long <- dat.long[order(dat.long$study, dat.long$time),]
rownames(dat.long) <- 1:nrow(dat.long)

### remove missing measurement occasions from dat.long
is.miss  <- is.na(dat.long$yi)
dat.long <- dat.long[!is.miss,]

### construct the full (block diagonal) V matrix with an AR(1) structure
rho.within <- .97 ### value as estimated by Ishak et al. (2007)
V <- lapply(split(with(dat, cbind(v1i, v2i, v3i, v4i)), dat$study), diag)
V <- lapply(V, function(v) sqrt(v) %*% toeplitz(ARMAacf(ar=rho.within, lag.max=3)) %*% sqrt(v))
V <- bldiag(V)
V <- V[!is.miss,!is.miss] ### remove missing measurement occasions from V

### plot data
with(dat.long, interaction.plot(time, study, yi, type="b", pch=19, lty="solid", xaxt="n",
                legend=FALSE, xlab="Time Point", ylab="Mean Difference", bty="l"))
axis(side=1, at=1:4, lab=c("1 (3 months)", "2 (6 months)", "3 (12 months)", "4 (12+ months)"))

### multivariate model with heteroscedastic AR(1) structure for the true effects
res <- rma.mv(yi, V, mods = ~ factor(time) - 1, random = ~ time | study,
              struct = "HAR", data = dat.long)
print(res, digits=2)
```

---

dat.konstantopoulos2011

*Studies on the Effects of Modified School Calendars on Student Achievement*

---

### Description

Results from 56 studies on the effects of modified school calendars on student achievement.

### Usage

```
dat.konstantopoulos2011
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **district** | numeric | district id number |
| **study** | numeric | study id number |
| **yi** | numeric | standardized mean difference |
| **vi** | numeric | corresponding sampling variance |
| **year** | numeric | year of the study |

**Details**

Instead of following the more traditional school calendar with a long summer break (in addition to a short winter and spring break), some schools have switched to a modified school calendar comprising more frequent but shorter intermittent breaks (e.g., 9 weeks of school followed by 3 weeks off), while keeping the total number of days at school approximately the same. The effects of using such a modified calendar on student achievement have been examined in a number of studies and were meta-analyzed by Cooper et al. (2003).

The dataset (taken from Konstantopoulos, 2011) contains the results from 56 studies, each comparing the level of academic achievement in a group of students following a modified school calendar with that of a group of students following a more traditional school calendar. The difference between the two groups was quantified in terms of a standardized mean difference (with positive values indicating a higher mean level of achievement in the group following the modified school calendar).

The studies were conducted at various schools that were clustered within districts. The data therefore have a multilevel structure, with studies nested within districts. A multilevel meta-analysis of these data can be used to estimate and account for the amount of heterogeneity between districts and between schools within districts.

**Source**

Konstantopoulos, S. (2011). Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods*, **2**, 61–76.

**References**

Cooper, H., Valentine, J. C., Charlton, K., & Melson, A. (2003). The effects of modified school calendars on student achievement and on school and community attitudes. *Review of Educational Research*, **73**, 1–52.

**Examples**

```
### load data
dat <- get(data(dat.konstantopoulos2011))

### regular random-effects model
res <- rma(yi, vi, data=dat)
print(res, digits=3)

### regular random-effects model using rma.mv()
res <- rma.mv(yi, vi, random = ~ 1 | study, data=dat)
print(res, digits=3)

### multilevel random-effects model
res.ml <- rma.mv(yi, vi, random = list(~ 1 | district, ~ 1 | study), data=dat)
print(res.ml, digits=3)

### profile variance components
par(mfrow=c(2,1))
profile(res.ml, sigma2=1)
```

```
profile(res.ml, sigma2=2)

### multivariate parameterization of the model
res.mv <- rma.mv(yi, vi, random = ~ factor(study) | district, data=dat)
print(res.mv, digits=3)

### tau^2 from multivariate model = sum of the two variance components from the multilevel model
round(sum(res.ml$sigma2), 3)

### rho from multivariate model = intraclass correlation coefficient based on the multilevel model
round(res.ml$sigma2[1] / sum(res.ml$sigma2), 3)
```

---

dat.lee2004                     *Studies on Acupoint P6 Stimulation for Preventing Nausea*

---

### Description

Results from studies examining the effectiveness of wrist acupuncture point P6 stimulation for preventing postoperative nausea.

### Usage

```
dat.lee2004
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **id** | numeric | trial id number |
| **study** | character | first author |
| **year** | numeric | study year |
| **ai** | numeric | number of patients experiencing nausea in the treatment group |
| **n1i** | numeric | total number of patients in treatment group |
| **ci** | numeric | number of patients experiencing nausea in the sham group |
| **n2i** | numeric | total number of patients in the sham group |

### Details

Postoperative nausea and vomiting are common complications following surgery and anaesthesia. As an alternative to drug therapy, acupuncture has been studied as a potential treatment in several trials. The dataset contains the results from 16 clinical trials examining the effectiveness of wrist acupuncture point P6 stimulation for preventing postoperative nausea.

### Source

Lee, A., & Done, M. L. (2004). Stimulation of the wrist acupuncture point P6 for preventing postoperative nausea and vomiting. *Cochrane Database of Systematic Reviews*, **3**, CD003281.

## Examples

```
### load data
dat <- get(data(dat.lee2004))

### meta-analysis based on relative risks
res <- rma(measure="RR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat)
res
predict(res, transf=exp, digits=2)
```

dat.linde2005 *Studies on the Effectiveness of St. John's Wort for Treating Depression*

## Description

Results from 26 studies on the effectiveness of Hypericum perforatum extracts (St. John's wort) for treating depression.

## Usage

```
dat.linde2005
```

## Format

The data frame contains the following columns:

| **id** | numeric | study number |
| **study** | character | study author(s) |
| **year** | numeric | publication year |
| **country** | character | study location |
| **ni** | numeric | total sample size |
| **major** | numeric | sample restricted to patients who met criteria for major depression |
| **baseline** | numeric | HRSD baseline score |
| **version** | numeric | HRSD version (17 or 21 items) |
| **duration** | numeric | study duration (in weeks) |
| **prep** | character | Hypericum extract preparation |
| **dosage** | numeric | doage (in mg) |
| **response** | numeric | definition of response (1 = HRSD score reduction of at least 50% or HRSDscore after therapy <10; |
| **ai** | numeric | number of responses in treatment group |
| **n1i** | numeric | number of patients in treatment group |
| **ci** | numeric | number of responses in placebo group |
| **n2i** | numeric | number of patients in placebo group |
| **group** | numeric | stratification variable used by the authors (1 = smaller trials restricted to major depression; 2 = large |

## Details

The dataset includes the results from 26 double-blind placebo-controlled trials on the effectiveness of Hypericum perforatum extracts (St. John's wort) for treating depression (note that 2 studies did

not provide sufficient response information).

Data were extracted from Table 1 and Figure 3. For study duration, the assessment week (instead of the total study duration) was coded for Philipp et al. (1999) and Montgomery et al. (2000). For dosage, the midpoint was coded when a range of values was given.

### Source

Linde, K., Berner, M., Egger, M., & Mulrow, C. (2005). St John's wort for depression: Meta-analysis of randomised controlled trials. *British Journal of Psychiatry*, **186**, 99–107.

### References

Viechtbauer, W. (2007). Accounting for heterogeneity via random-effects models and moderator analyses in meta-analysis. *Zeitschrift für Psychologie / Journal of Psychology*, **215**, 104–121.

### Examples

```
### load data
dat <- get(data(dat.linde2005))

### remove studies with no response information and study with no responses in either group
dat <- dat[-c(5,6,26),]

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=ai, ci=ci, n1i=n1i, n2i=n2i, data=dat)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(yi, vi, data=dat, method="DL")
res

### mixed-effects meta-regression model with stratification variable
res <- rma(yi, vi, mods = ~ factor(group) - 1, data=dat, method="DL")
res

### predicted average relative risk for each level of the stratification variable
predict(res, newmods=diag(4), transf=exp, digits=2)
```

---

dat.mcdaniel1994          *Studies on the Validity of Employment Interviews*

---

### Description

Results from 160 studies on the correlation between employment interview assessments and job performance.

### Usage

```
dat.mcdaniel1994
```

**Format**

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **ni** | numeric | sample size of the study |
| **ri** | numeric | observed correlation |
| **type** | character | interview type (j = job-related, s = situational, p = psychological) |
| **struct** | character | interview structure (u = unstructured, s = structured) |

**Details**

The 160 studies provide data in terms of the correlation between employment interview performance and actual job performance. In addition, the interview type and the interview structure are indicated.

McDaniel et al. (1994) describe the interview type and structure variables as follows. "Questions in situational interviews [...] focus on the individual's ability to project what his or her behavior would be in a given situation. [...] Job-related interviews are those in which the interviewer is a personnel officer or hiring authority and the questions attempt to assess past behaviors and job-related information, but most questions are not considered situational. Psychological interviews are conducted by a psychologist, and the questions are intended to assess personal traits, such as dependability." In structured interviews, "the questions and acceptable responses were specified in advance and the responses were rated for appropriateness of content. [...] Unstructured interviews gather applicant information in a less systematic manner than do structured interviews. Although the questions may be specified in advance, they usually are not, and there is seldom a formalized scoring guide. Also, all persons being interviewed are not typically asked the same questions."

The goal of the meta-analysis was to examine the overall criterion-related validity of employment interviews and to examine whether the validity depends on the type and structure of the interview.

The data in this dataset were obtained from Table A.2 in Rothstein, Sutton, and Borenstein (2005, p. 325-329). Note that the type and struct variables contain some NAs.

**Source**

Rothstein, H. R., Sutton, A. J., & Borenstein, M. (Eds.). (2005). *Publication bias in meta-analysis: Prevention, assessment, and adjustments*. Chichester, England: Wiley.

**References**

McDaniel, M. A., Whetzel, D. L., Schmidt, F. L., & Maurer, S. D. (1994). The validity of employment interviews: A comprehensive review and meta-analysis. *Journal of Applied Psychology*, **79**, 599–616.

**Examples**

```
### load data
dat <- get(data(dat.mcdaniel1994))

### calculate r-to-z transformed correlations and corresponding sampling variances
dat <- escalc(measure="ZCOR", ri=ri, ni=ni, data=dat)
```

```
dat

### meta-analysis of the transformed correlations using a random-effects model
res <- rma(yi, vi, data=dat)
res

### average correlation with 95% CI
predict(res, transf=transf.ztor)

### mixed-effects model with interview type as factor
### note: job-related interviews is the reference level
rma(yi, vi, mods = ~ factor(type), data=dat)

### mixed-effects model with interview structure as factor
### note: structured interviews is the reference level
rma(yi, vi, mods = ~ factor(struct), data=dat)

### note: the interpretation of the results is difficult since all
### situational interviews were structured, almost all psychological
### interviews were unstructured, and actually for the majority of
### the psychological interviews it was unknown whether the interview
### was structured or unstructured
table(dat$type, dat$struct, useNA="always")

### meta-analysis of raw correlations using a random-effects model
res <- rma(measure="COR", ri=ri, ni=ni, data=dat.mcdaniel1994)
res
```

---

dat.molloy2014 *Studies on the Relationship between Conscientiousness and Medication Adherence*

---

### Description

Results from 16 studies on the correlation between conscientiousness and medication adherence.

### Usage

```
dat.molloy2014
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **authors** | character | study authors |
| **year** | numeric | publication year |
| **ni** | numeric | sample size of the study |
| **ri** | numeric | observed correlation |
| **controls** | character | number of variables controlled for |
| **design** | character | whether a cross-sectional or prospective design was used |

| **a_measure** | character | type of adherence measure (self-report or other) |
| **c_measure** | character | type of conscientiousness measure (NEO or other) |
| **meanage** | numeric | mean age of the sample |
| **quality** | numeric | methodological quality |

## Details

Conscientiousness, one of the big-5 personality traits, can be defined as "socially prescribed impulse control that facilitates task- and goal-directed behaviour, such as thinking before acting, delaying gratification, following norms and rules and planning, organising and prioritising tasks" (John & Srivastava, 1999). Conscientiousness has been shown to be related to a number of health-related behaviors (e.g., tobacco/alcohol/drug use, diet and activity patterns, risky behaviors). A recent meta-analysis by Molloy et al. (2014) examined to what extent conscientiousness is related to medication adherence, that is, the extent to which (typically chronically ill) patients follow a prescribed medication regimen (e.g., taking a daily dose of a cholesterol lowering drug in patients with high LDL serum cholesterol levels). The results from the 16 studies included in this meta-analysis are provided in this dataset.

Variable `a_measure` indicates whether adherence was measured based on self-reports or a more 'objective' measure (e.g., electronic monitoring of pill bottle openings, pill counts). Variable `c_measure` indicates whether conscientiousness was measured with some version of the NEO personality inventory or some other scale. Methodological quality was scored by the authors on a 1 to 4 scale with higher scores indicating higher quality (see article for details on how this score was derived).

## Source

Molloy, G. J., O'Carroll, R. E., & Ferguson, E. (2014). Conscientiousness and medication adherence: A meta-analysis. *Annals of Behavioral Medicine, 47,* 92–101.

## References

John, O. P., & Srivastava, S. (1999). The Big Five Trait taxonomy: History, measurement, and theoretical perspectives. In L. A. Pervin & O. P. John (Eds.), *Handbook of personality: Theory and research* (2nd ed., pp. 102-138). New York: Guilford Press.

## Examples

```
### load data
dat <- get(data(dat.molloy2014))

### calculate r-to-z transformed correlations and corresponding sampling variances
dat <- escalc(measure="ZCOR", ri=ri, ni=ni, data=dat, slab=paste(authors, year, sep=", "))
dat

### meta-analysis of the transformed correlations using a random-effects model
res <- rma(yi, vi, data=dat)
res

### average correlation with 95% CI
predict(res, digits=3, transf=transf.ztor)
```

```
### forest plot
forest(res, addcred=TRUE, xlim=c(-1.6,1.6), atransf=transf.ztor,
       at=transf.rtoz(c(-.4,-.2,0,.2,.4,.6)), digits=c(2,1))
text(-1.6, 18, "Author(s), Year", pos=4)
text( 1.6, 18, "Correlation [95% CI]", pos=2)

### funnel plot
funnel(res)
```

---

dat.nielweise2007      *Studies on Anti-Infective-Treated Central Venous Catheters for Prevention of Catheter-Related Bloodstream Infections*

---

## Description

Results from 18 studies comparing the risk of catheter-related bloodstream infection when using anti-infective-treated versus standard catheters in the acute care setting.

## Usage

```
dat.nielweise2007
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **author** | character | (first) author |
| **year** | numeric | publication year |
| **ai** | numeric | number of CRBSIs in patients receiving an anti-infective catheter |
| **n1i** | numeric | number of patients receiving an anti-infective catheter |
| **ci** | numeric | number of CRBSIs in patients receiving a standard catheter |
| **n2i** | numeric | number of patients receiving a standard catheter |

## Details

The use of a central venous catheter may lead to a catheter-related bloodstream infection (CRBSI), which in turn increases the risk of morbidity and mortality. Anti-infective-treated catheters have been developed that are meant to reduce the risk of CRBSIs. Niel-Weise et al. (2007) conducted a meta-analysis of studies comparing infection risk when using anti-infective-treated versus standard catheters in the acute care setting. The results from 18 such studies are included in this dataset.

The dataset was used in the article by Stijnen et al. (2010) to illustrate various generalized linear mixed-effects models for the meta-analysis of proportions and odds ratios (see 'References').

## Source

Niel-Weise, B. S., Stijnen, T., & van den Broek, P. J. (2007). Anti-infective-treated central venous catheters: A systematic review of randomized controlled trials. *Intensive Care Medicine*, **33**, 2058–2068.

## References

Stijnen, T., Hamza, T. H., & Ozdemir, P. (2010). Random effects meta-analysis of event outcome in the framework of the generalized linear mixed model with applications in sparse data. *Statistics in Medicine*, **29**, 3046–3067.

## Examples

```
### load data
dat <- get(data(dat.nielweise2007))

### standard (inverse-variance) random-effects model
res <- rma(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat, drop00=TRUE)
print(res, digits=3)
predict(res, transf=exp, digits=2)

### random-effects conditional logistic model
## Not run:
res <- rma.glmm(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat, model="CM.EL")
print(res, digits=3)
predict(res, transf=exp, digits=2)

## End(Not run)
```

---

| dat.nielweise2008 | *Studies on Anti-Infective-Treated Central Venous Catheters for Prevention of Catheter-Related Bloodstream Infections* |
|---|---|

---

## Description

Results from 18 studies comparing the risk of catheter-related bloodstream infection when using anti-infective-treated versus standard catheters for total parenteral nutrition or chemotherapy.

## Usage

```
dat.nielweise2008
```

## Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **authors** | character | study authors |

| **year** | numeric | publication year |
|---|---|---|
| **x1i** | numeric | number of CRBSIs in patients receiving an anti-infective catheter |
| **t1i** | numeric | total number of catheter days for patients receiving an anti-infective catheter |
| **x2i** | numeric | number of CRBSIs in patients receiving a standard catheter |
| **t2i** | numeric | total number of catheter days for patients receiving a standard catheter |

### Details

The use of a central venous catheter may lead to a catheter-related bloodstream infection (CRBSI), which in turn increases the risk of morbidity and mortality. Anti-infective-treated catheters have been developed that are meant to reduce the risk of CRBSIs. Niel-Weise et al. (2008) conducted a meta-analysis of studies comparing infection risk when using anti-infective-treated versus standard catheters for total parenteral nutrition or chemotherapy. The results from 9 such studies are included in this dataset.

The dataset was used in the article by Stijnen et al. (2010) to illustrate various generalized linear mixed-effects models for the meta-analysis of incidence rates and incidence rate ratios (see 'References').

### Source

Niel-Weise, B. S., Stijnen, T., & van den Broek, P. J. (2008). Anti-infective-treated central venous catheters for total parenteral nutrition or chemotherapy: A systematic review. *Journal of Hospital Infection*, **69**, 114–123.

### References

Stijnen, T., Hamza, T. H., & Ozdemir, P. (2010). Random effects meta-analysis of event outcome in the framework of the generalized linear mixed model with applications in sparse data. *Statistics in Medicine*, **29**, 3046–3067.

### Examples

```
### load data
dat <- get(data(dat.nielweise2008))

### standard (inverse-variance) random-effects model
res <- rma(measure="IRR", x1i=x1i, t1i=t1i, x2i=x2i, t2i=t2i, data=dat)
print(res, digits=3)
predict(res, transf=exp, digits=2)

### random-effects conditional Poisson model
## Not run:
res <- rma.glmm(measure="IRR", x1i=x1i, t1i=t1i, x2i=x2i, t2i=t2i, data=dat, model="CM.EL")
print(res, digits=3)
predict(res, transf=exp, digits=2)

## End(Not run)
```

---

dat.normand1999                 *Studies on the Length of Hospital Stay of Stroke Patients*

---

**Description**

Results from 9 studies on the length of the hospital stay of stroke patients under specialized care
and under conventional/routine (non-specialist) care.

**Usage**

```
dat.normand1999
```

**Format**

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **source** | character | source of data |
| **n1i** | numeric | number of patients under specialized care |
| **m1i** | numeric | mean length of stay (in days) under specialized care |
| **sd1i** | numeric | standard deviation of the length of stay under specialized care |
| **n2i** | numeric | number of patients under routine care |
| **m2i** | numeric | mean length of stay (in days) under routine care |
| **sd2i** | numeric | standard deviation of the length of stay under routine care |

**Details**

The 9 studies provide data in terms of the mean length of the hospital stay (in days) of stroke
patients under specialized care and under conventional/routine (non-specialist) care. The goal of
the meta-analysis was to examine the hypothesis whether specialist stroke unit care will result in a
shorter length of hospitalization compared to routine management.

**Source**

Normand, S. T. (1999). Meta-analysis: Formulating, evaluating, combining, and reporting. *Statistics in Medicine*, **18**, 321–359.

**Examples**

```
### load data
dat <- get(data(dat.normand1999))

### calculate mean differences and corresponding sampling variances
dat <- escalc(measure="MD", m1i=m1i, sd1i=sd1i, n1i=n1i, m2i=m2i, sd2i=sd2i, n2i=n2i, data=dat)
dat

### meta-analysis of mean differences using a random-effects model
res <- rma(yi, vi, data=dat)
```

```
res

### meta-analysis of standardized mean differences using a random-effects model
res <- rma(measure="SMD", m1i=m1i, sd1i=sd1i, n1i=n1i, m2i=m2i, sd2i=sd2i, n2i=n2i,
           data=dat, slab=source)
res

### draw forest plot
forest(res, xlim=c(-7,5), alim=c(-3,1))
text(-7, 11, "Study/Source",          pos=4)
text( 5, 11, "Observed SMD [95% CI]", pos=2)
```

---

| dat.pignon2000 | *Studies on the Effectiveness of Locoregional Treatment plus Chemotherapy for Head and Neck Squamous-Cell Carcinoma* |
|---|---|

---

## Description

Results from studies examining mortality risk in patients with nonmetastatic head and neck squamous-cell carcinoma receiving either locoregional treatment plus chemotherapy versus locoregional treatment alone.

## Usage

```
dat.pignon2000
```

## Format

The data frame contains the following columns:

| **id** | numeric | study id number |
|---|---|---|
| **trial** | character | trial abbreviation |
| **OmE** | numeric | observed minus expected number of deaths in the locoregional treatment plus chemotherapy group |
| **V** | numeric | corresponding variance |
| **grp** | numeric | timing of chemotherapy: 1 = adjuvant, 2 = neoadjuvant, 3 = concomitant |

## Details

The purpose of this meta-analysis was to examine the mortality risk in patients with nonmetastatic head and neck squamous-cell carcinoma receiving either locoregional treatment plus chemotherapy versus locoregional treatment alone. For 65 trials, the dataset provides the observed minus expected number of deaths and corresponding variances in the locoregional treatment plus chemotherapy group. Based on these values, we can estimate the log hazard ratios with `OmE/V` and the corresponding sampling variance with `1/V`.

The trials were also divided according to the timing of the chomotherapy: (1) adjuvant, after the locoregional treatment, (2) neoadjuvant, before the locoregional treatment, and (3) concomitant, chemotherapy given concomitantly or alternating with radiotherapy.

**Source**

Pignon, J. P., Bourhis, J., Domenge, C., & Designe, L. (2000). Chemotherapy added to locoregional treatment for head and neck squamous-cell carcinoma: Three meta-analyses of updated individual data. *Lancet*, **355**(9208), 949–955.

**Examples**

```
### load data
dat <- get(data(dat.pignon2000))

### compute estimated log hazard ratios and sampling variances
dat$yi <- with(dat, OmE/V)
dat$vi <- with(dat, 1/V)

### meta-analysis based on all 65 trials
res <- rma(yi, vi, data=dat, method="FE", digits=2)
res
predict(res, transf=exp)

### only adjuvant trials
res <- rma(yi, vi, data=dat, method="FE", subset=grp==1, digits=2)
res
predict(res, transf=exp)

### only neoadjuvant trials
res <- rma(yi, vi, data=dat, method="FE", subset=grp==2, digits=2)
res
predict(res, transf=exp)

### only concomitant trials
res <- rma(yi, vi, data=dat, method="FE", subset=grp==3, digits=2)
res
predict(res, transf=exp)
```

---

dat.pritz1997          *Studies on the Effectiveness of Hyperdynamic Therapy for Treating Cerebral Vasospasm*

---

**Description**

Results from 14 studies on the effectiveness of hyperdynamic therapy for treating cerebral vasospasm.

**Usage**

```
dat.pritz1997
```

**Format**

The data frame contains the following columns:

| **study** | numeric | study number |
|---|---|---|
| **authors** | character | study authors |
| **xi** | numeric | number of patients that improved with hyperdynamic therapy |
| **ni** | numeric | total number of patients treated |

## Details

As described in Zhou et al. (1999), "hyperdynamic therapy refers to induced hypertension and hypervolaemia (volume expansion) to treat ischaemic symptoms due to vasospasm, and the success of this therapy is defined as clinical improvement in terms of neurologic deficits." For each study that was included in the meta-analysis, the dataset includes information on the number of patients that improved under this form of therapy and the total number of patients that were treated. The goal of the meta-analysis is to estimate the true (average) success rate of hyperdynamic therapy.

## Source

Zhou, X.-H., Brizendine, E. J., & Pritz, M. B. (1999). Methods for combining rates from several studies. *Statistics in Medicine*, **18**, 557–566.

## References

Pritz, M. B. (1997). Treatment of cerebral vasospasm due to aneurysmal subarachnoid hemorrhage: Past, present, and future of hyperdynamic therapy. *Neurosurgery Quarterly*, **7**, 273–285.

## Examples

```
### load data
dat <- get(data(dat.pritz1997))

### computation of "weighted average" in Zhou et al. (1999), Table IV
dat <- escalc(measure="PR", xi=xi, ni=ni, data=dat, add=0)
theta.hat    <- sum(dat$ni * dat$yi) / sum(dat$ni)
se.theta.hat <- sqrt(sum(dat$ni^2 * dat$vi) / sum(dat$ni)^2)
ci.lb        <- theta.hat - 1.96*se.theta.hat
ci.ub        <- theta.hat + 1.96*se.theta.hat
round(c(estimate = theta.hat, se = se.theta.hat, ci.lb = ci.lb, ci.ub = ci.ub), 4)

### random-effects model with raw proportions
dat <- escalc(measure="PR", xi=xi, ni=ni, data=dat)
res <- rma(yi, vi, data=dat)
predict(res)

### random-effects model with logit transformed proportions
dat <- escalc(measure="PLO", xi=xi, ni=ni, data=dat)
res <- rma(yi, vi, data=dat)
predict(res, transf=transf.ilogit)
```

---

dat.raudenbush1985          *Studies on Assessing the Effects of Teacher Expectations on Pupil IQ*

---

### Description

Results from 19 studies examining how teachers' expectations about their pupils can influence actual IQ levels.

### Usage

```
dat.raudenbush1985
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **author** | character | study author(s) |
| **year** | numeric | publication year |
| **weeks** | numeric | weeks of contact prior to expectancy induction |
| **setting** | character | whether tests were group or individually administered |
| **tester** | character | whether test administrator was aware or blind |
| **yi** | numeric | standardized mean difference |
| **vi** | numeric | corresponding sampling variance |

### Details

In the so-called 'Pygmalion study' (Rosenthal & Jacobson, 1968), "all of the predominantly poor children in the so-called Oak elementary school were administered a test pretentiously labeled the 'Harvard Test of Inflected Acquisition.' After explaining that this newly designed instrument had identified those children most likely to show dramatic intellectual growth during the coming year, the experimenters gave tne names of these 'bloomers' to the teachers. In truth, the test was a traditional IQ test and the 'bloomers' were a randomly selected 20% of the student population. After retesting the children 8 months later, the experimenters reported that those predicted to bloom had in fact gained significantly more in total IQ (nearly 4 points) and reasoning IQ (7 points) than the control group children. Further, at the end of the study, the teachers rated the experimental children as intellectually more curious, happier, better adjusted, and less in need of approval than their control group peers" (Raudenbush, 1984).

In the following years, a series of studies were conducted attempting to replicate this rather controversial finding. However, the great majority of those studies were unable to demonstrate a statistically significant difference between the two experimental groups in terms of IQ scores. Raudenbush (1984) conducted a meta-analysis based on 19 such studies to further examine the evidence for the existence of the 'Pygmalion effect'. The dataset includes the results from these studies.

The effect size measure used for the meta-analysis was the standardized mean difference (yi), with positive values indicating that the supposed 'bloomers' had, on average, higher IQ scores than those

in the control group. The weeks variable indicates the number of weeks of prior contact between teachers and students before the expectancy induction. Testing was done either in a group setting or individually, which is indicated by the setting variable. Finally, the tester variable indicates whether the test administrators were either aware or blind to the researcher-provided designations of the children's intellectual potential.

The data in this dataset were obtained from Raudenbush and Bryk (1985) with information on the setting and tester variables extracted from Raudenbush (1984).

### Source

Raudenbush, S. W. (1984). Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 18 experiments. *Journal of Educational Psychology*, **76**, 85–97.

Raudenbush, S. W., & Bryk, A. S. (1985). Empirical Bayes meta-analysis. *Journal of Educational Statistics*, **10**, 75–98.

### Examples

```
### load data
dat <- get(data(dat.raudenbush1985))

### random-effects model
res <- rma(yi, vi, data=dat)
res
```

---

dat.riley2003                    *Studies on MYC-N as a Prognostic Marker for Neuroblastoma*

---

### Description

Results from 81 studies examining overall and disease-free survival in neuroblastoma patients with amplified versus normal MYC-N protein levels.

### Usage

```
dat.riley2003
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **study** | numeric | study number |
| **yi** | numeric | log hazard ratio of the outcome in those with amplified versus normal MYC-N protein levels |
| **sei** | numeric | standard error of the log hazard ratio |
| **outcome** | character | outcome (OS = overall survival; DFS = disease-free survival) |

**Details**

The meta-analysis by Riley et al. (2003) examined a variety of prognostic markers for overall and disease-free survival in patients with neuroblastoma. One of the markers examined was amplified levels of the MYC-N protein, with is associated with poorer outcomes.

The dataset given here was extracted from Riley (2011) and has been used in several other publications (e.g., Riley et al., 2004, 2007). The dataset provides the (log) hazard ratios (and corresponding standard errors) with respect to these two outcomes in 81 studies, with positive values indicating a greater risk of death (for OS) or disease recurrence/death (for DFS) for patients with high MYC-N levels compared to those with normal/low levels. Note that information on both outcomes could only be extracted from 17 studies (39 studies only provided sufficient information to extract the OS estimate, while 25 studies only allowed for extraction of the DFS estimate).

**Source**

Riley, R. D., Sutton, A. J., Abrams, K. R., & Lambert, P. C. (2004). Sensitivity analyses allowed more appropriate and reliable meta-analysis conclusions for multiple outcomes when missing data was present. *Journal of Clinical Epidemiology*, **57**, 911–924.

Riley, R. D., Abrams, K. R., Lambert, P. C., Sutton, A. J., & Thompson, J. R. (2007). An evaluation of bivariate random-effects meta-analysis for the joint synthesis of two correlated outcomes. *Statistics in Medicine*, **26**, 78–97.

Riley, R. D. (2011). Erratum. *Statistics in Medicine*, **30**, 400.

**References**

Riley, R. D., Burchill, S. A., Abrams, K. R., Heney, D., Lambert, P. C., Jones, D. R., et al. (2003). A systematic review and evaluation of the use of tumour markers in paediatric oncology: Ewing's sarcoma and neuroblastoma. *Health Technology Assessment*, **7**, 1–162.

**Examples**

```
### load data
dat <- get(data(dat.riley2003))

### random-effects model analysis for outcome DFS
res <- rma(yi, sei=sei, data=dat, subset=(outcome == "DFS"), method="DL")
res
predict(res, transf=exp, digits=2)

### random-effects model analysis for outcome OS
res <- rma(yi, sei=sei, data=dat, subset=(outcome == "OS"), method="DL")
res
predict(res, transf=exp, digits=2)
```

---

**dat.yusuf1985**               *Studies of Beta Blockers During and After Myocardial Infarction*

---

### Description

Results from studies examining the effectiveness of beta blockers for reducing mortality and reinfarction.

### Usage

```
dat.yusuf1985
```

### Format

The data frame contains the following columns:

| | | |
|---|---|---|
| **table** | character | table number |
| **id** | character | trial id number |
| **trial** | character | trial name or first author |
| **ai** | numeric | number of deaths/reinfarctions in treatment group |
| **n1i** | numeric | number of patients in treatment group |
| **ci** | numeric | number of deaths/reinfarctions in control group |
| **n2i** | numeric | number of patients in control group |

### Details

The dataset contains table 6 (total mortality from short-term trials of oral beta blockers), 9 (total mortality at one week from trials with an initial IV dose of a beta blocker), 10 (total mortality from long-term trials with treatment starting late and mortality from day 8 onwards in long-term trials that began early and continued after discharge), 11 (nonfatal reinfarction from long-term trials of beta blockers), 12a (sudden death in long-term beta blocker trials), and 12b (nonsudden death in long-term beta blocker trials) from the meta-analysis by Yusuf et al. (1985) on the effectiveness of of beta blockers for reducing mortality and reinfarction.

The article also describes what is sometimes called Peto's one-step method for meta-analyzing $2 \times 2$ table data. This method is implemented in the rma.peto function.

### Source

Yusuf, S., Peto, R., Lewis, J., Collins, R., & Sleight, P. (1985). Beta blockade during and after myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Disease*, **27**, 335–371.

### Examples

```
### load data
dat <- get(data(dat.yusuf1985))
```

```
### to select a table for the analysis
tab <- "6" ### either: 6, 9, 10, 11, 12a, 12b

### to double-check total counts as reported in article
apply(dat[dat$table==tab,4:7], 2, sum, na.rm=TRUE)

### meta-analysis using Peto's one-step method
res <- rma.peto(ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat, subset=(table==tab))
res
predict(res, transf=exp, digits=2)
```

---

escalc                          *Calculate Effect Sizes and Outcome Measures*

---

#### Description

The function can be used to calculate various effect sizes or outcome measures (and the correspond-
ing sampling variances) that are commonly used in meta-analyses.

#### Usage

```
escalc(measure, formula, ...)

## Default S3 method:
escalc(measure, formula, ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i,
       m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, sdi, ni, data, slab, subset,
       add=1/2, to="only0", drop00=FALSE, vtype="LS",
       var.names=c("yi","vi"), add.measure=FALSE,
       append=TRUE, replace=TRUE, digits=4, ...)

## S3 method for class 'formula'
escalc(measure, formula, weights, data,
       add=1/2, to="only0", drop00=FALSE, vtype="LS",
       var.names=c("yi","vi"), digits=4, ...)
```

#### Arguments

measure      a character string indicating which effect size or outcome measure should be
             calculated. See 'Details' for possible options and how the data should be speci-
             fied.

formula      when using the formula interface of the function (see 'Details' below), a model
             formula specifying the data structure should be specified via this argument.
             When not using the formula interface, this argument should be ignored and the
             data required to calculate the effect sizes or outcomes are then passed to the
             function via the following set of arguments. See 'Details'.

weights      vector of weights to specify the group sizes or cell frequencies (only needed
             when using the formula interface). See 'Details'.

| ai | vector to specify the $2 \times 2$ table frequencies (upper left cell). |
|---|---|
| bi | vector to specify the $2 \times 2$ table frequencies (upper right cell). |
| ci | vector to specify the $2 \times 2$ table frequencies (lower left cell). |
| di | vector to specify the $2 \times 2$ table frequencies (lower right cell). |
| n1i | vector to specify the group sizes or row totals (first group/row). |
| n2i | vector to specify the group sizes or row totals (second group/row). |
| x1i | vector to specify the number of events (first group). |
| x2i | vector to specify the number of events (second group). |
| t1i | vector to specify the total person-times (first group). |
| t2i | vector to specify the total person-times (second group). |
| m1i | vector to specify the means (first group or time point). |
| m2i | vector to specify the means (second group or time point). |
| sd1i | vector to specify the standard deviations (first group or time point). |
| sd2i | vector to specify the standard deviations (second group or time point). |
| xi | vector to specify the frequencies of the event of interest. |
| mi | vector to specify the frequencies of the complement of the event of interest or the group means. |
| ri | vector to specify the raw correlation coefficients. |
| ti | vector to specify the total person-times. |
| sdi | vector to specify the standard deviations. |
| ni | vector to specify the sample/group sizes. |
| data | optional data frame containing the variables given to the arguments above. |
| slab | optional vector with labels for the studies. |
| subset | optional vector indicating the subset of studies that should be used. This can be a logical vector or a numeric vector indicating the indices of the studies to include. |
| add | a non-negative number indicating the amount to add to zero cells, counts, or frequencies. See 'Details'. |
| to | a character string indicating when the values under add should be added (either "all", "only0", "if0all", or "none"). See 'Details'. |
| drop00 | logical indicating whether studies with no cases/events (or only cases) in both groups should be dropped when calculating the observed outcomes of the individual studies. See 'Details'. |
| vtype | a character string indicating the type of sampling variances to calculate (either "LS", "UB", "ST", or vtype="CS"). See 'Details'. |
| var.names | a character string with two elements, specifying the name of the variable for the observed outcomes and the name of the variable for the corresponding sampling variances (default is "yi" and "vi"). |

add.measure     logical indicating whether a variable should be added to the data frame (with
                default name "measure") that indicates the type of outcome measure computed.
                When using this option, var.names can have a third element to change this
                variable name.

append          logical indicating whether the data frame specified via the data argument (if one
                has been specified) should be returned together with the observed outcomes and
                corresponding sampling variances (default is TRUE).

replace         logical indicating whether existing values for yi and vi in the data frame should
                be replaced or not. Only relevant when append=TRUE and the data frame al-
                ready contains the yi and vi variables. If replace=TRUE (the default), all of the
                existing values will be overwritten. If replace=FALSE, only NA values will be
                replaced. See 'Value' section below for more details.

digits          integer specifying the number of decimal places to which the printed results
                should be rounded (default is 4). Note that the values are stored without round-
                ing in the returned object.

...             other arguments.

**Details**

Before a meta-analysis can be conducted, the relevant results from each study must be quantified
in such a way that the resulting values can be further aggregated and compared. Depending on (a)
the goals of the meta-analysis, (b) the design and types of studies included, and (c) the informa-
tion provided therein, one of the various effect size or outcome measures described below may be
appropriate for the meta-analysis and can be computed with the escalc function.

The measure argument is a character string specifying which outcome measure should be calcu-
lated (see below for the various options), arguments ai through ni are then used to specify the
information needed to calculate the various measures (depending on the chosen outcome measure,
different arguments need to be specified), and data can be used to specify a data frame containing
the variables given to the previous arguments. The add, to, and drop00 arguments may be needed
when dealing with frequency or count data that may need special handling when some of the fre-
quencies or counts are equal to zero (see below for details). Finally, the vtype argument is used to
specify how to estimate the sampling variances (again, see below for details).

To provide a structure to the various effect size or outcome measures that can be calculated with the
escalc function, we can distinguish between measures that are used to:

  • contrast two (either experimentally created or naturally occuring) groups,

  • describe the direction and strength of the association between two variables,

  • summarize some characteristic or attribute of individual groups.

Furthermore, where appropriate, we can further distinguish between measures that are applicable
when the characteristic, response, or dependent variable assessed in the individual studies is:

  • a dichotomous (binary) variable (e.g., remission versus no remission),

  • a count of events per time unit (e.g., number of migraines per year),

  • a quantitative variable (e.g., amount of depression as assessed by a rating scale).

**Outcome Measures for Two-Group Comparisons:**

In many meta-analyses, the goal is to synthesize the results from studies that compare or contrast two groups. The groups may be experimentally defined (e.g., a treatment and a control group created via random assignment) or may naturally occur (e.g., men and women, employees working under high- versus low-stress conditions, people exposed to some environmental risk factor versus those not exposed).

*Measures for Dichotomous Variables:*

In various fields (such as the health and medical sciences), the response or outcome variable measured is often dichotomous (binary), so that the data from a study comparing two different groups can be expressed in terms of a $2 \times 2$ table, such as:

|         | outcome 1 | outcome 2 | total |
|---------|-----------|-----------|-------|
| group 1 | ai        | bi        | n1i   |
| group 2 | ci        | di        | n2i   |

where `ai`, `bi`, `ci`, and `di` denote the cell frequencies (i.e., the number of people falling into a particular category) and `n1i` and `n2i` the row totals (i.e., the group sizes).

For example, in a set of randomized clinical trials, group 1 and group 2 may refer to the treatment and placebo/control group, respectively, with outcome 1 denoting some event of interest (e.g., death, complications, failure to improve under the treatment) and outcome 2 its complement. Similarly, in a set of cohort studies, group 1 and group 2 may denote those who engage in and those who do not engage in a potentially harmful behavior (e.g., smoking), with outcome 1 denoting the development of a particular disease (e.g., lung cancer) during the follow-up period. Finally, in a set of case-control studies, group 1 and group 2 may refer to those with the disease (i.e., cases) and those free of the disease (i.e., controls), with outcome 1 denoting, for example, exposure to some risk environmental risk factor in the past and outcome 2 non-exposure. Note that in all of these examples, the stratified sampling scheme fixes the row totals (i.e., the group sizes) by design.

A meta-analysis of studies reporting results in terms of $2 \times 2$ tables can be based on one of several different outcome measures, including the relative risk (risk ratio), the odds ratio, the risk difference, and the arcsine square-root transformed risk difference (e.g., Fleiss & Berlin, 2009, Rücker et al., 2009). For any of these outcome measures, one needs to specify the cell frequencies via the `ai`, `bi`, `ci`, and `di` arguments (or alternatively, one can use the `ai`, `ci`, `n1i`, and `n2i` arguments).

The options for the `measure` argument are then:

- `"RR"` for the *log relative risk*.
- `"OR"` for the *log odds ratio*.
- `"RD"` for the *risk difference*.
- `"AS"` for the *arcsine square-root transformed risk difference* (Rücker et al., 2009).
- `"PETO"` for the *log odds ratio* estimated with Peto's method (Yusuf et al., 1985).

Note that the log is taken of the relative risk and the odds ratio, which makes these outcome measures symmetric around 0 and yields corresponding sampling distributions that are closer to normality.

Cell entries with a zero count can be problematic, especially for the relative risk and the odds ratio. Adding a small constant to the cells of the $2 \times 2$ tables is a common solution to this problem. When `to="only0"` (the default), the value of `add` (the default is 1/2) is added to each cell of those $2 \times 2$ tables with at least one cell equal to 0. When `to="all"`, the value of `add` is

added to each cell of all $2 \times 2$ tables. When `to="if0all"`, the value of `add` is added to each cell of all $2 \times 2$ tables, but only when there is at least one $2 \times 2$ table with a zero cell. Setting `to="none"` or `add=0` has the same effect: No adjustment to the observed table frequencies is made. Depending on the outcome measure and the data, this may lead to division by zero inside of the function (when this occurs, the resulting value is recoded to `NA`). Also, studies where `ai=ci=0` or `bi=di=0` may be considered to be uninformative about the size of the effect and dropping such studies has sometimes been recommended (Higgins & Green, 2008). This can be done by setting `drop00=TRUE`. The values for such studies will then be set to `NA`.

A dataset corresponding to data of this type is provided in `dat.bcg`.

Assuming that the dichotomous outcome is actually a dichotomized version of the responses on an underlying quantitative scale, it is also possible to estimate the standardized mean difference based on $2 \times 2$ table data, using either the probit transformed risk difference or a transformation of the odds ratio (e.g., Cox & Snell, 1989; Chinn, 2000; Hasselblad & Hedges, 1995; Sanchez-Meca et al., 2003). The options for the `measure` argument are then:

- `"PBIT"` for the *probit transformed risk difference* as an estimate of the standardized mean difference.
- `"OR2DN"` for the *transformed odds ratio* as an estimate of the standardized mean difference (normal distributions).
- `"OR2DL"` for the *transformed odds ratio* as an estimate of the standardized mean difference (logistic distributions).

The probit transformation assumes that the responses on the underlying quantitative scale are normally distributed. There are two versions of the odds ratio transformation, the first also assuming normal distributions within the two groups, while the second assumes that the responses follow logistic distributions.

A dataset corresponding to data of this type is provided in `dat.gibson2002`.

*Measures for Event Counts:*
In medical and epidemiological studies comparing two different groups (e.g., treated versus untreated patients, exposed versus unexposed individuals), results are sometimes reported in terms of event counts (i.e., the number of events, such as strokes or myocardial infarctions) over a certain period of time. Data of this type are also referred to as 'person-time data'. In particular, assume that the studies report data in the form:

|         | number of events | total person-time |
|---------|:----------------:|:-----------------:|
| group 1 | x1i              | t1i               |
| group 2 | x2i              | t2i               |

where `x1i` and `x2i` denote the total number of events in the first and the second group, respectively, and `t1i` and `t2i` the corresponding total person-times at risk. Often, the person-time is measured in years, so that `t1i` and `t2i` denote the total number of follow-up years in the two groups.

This form of data is fundamentally different from what was described in the previous section, since the total follow-up time may differ even for groups of the same size and the individuals studied may experience the event of interest multiple times. Hence, different outcome measures than the ones described in the previous section must be considered when data are reported in this format. These inlude the incidence rate ratio, the incidence rate difference, and the square-root transformed incidence rate difference (Bagos & Nikolopoulos, 2009; Rothman et al., 2008). For any of these outcome measures, one needs to specify the total number of events via the `x1i` and `x2i` arguments and the corresponding total person-time values via the `t1i` and `t2i` arguments.

The options for the `measure` argument are then:

- `"IRR"` for the *log incidence rate ratio*.
- `"IRD"` for the *incidence rate difference*.
- `"IRSD"` for the *square-root transformed incidence rate difference*.

Note that the log is taken of the incidence rate ratio, which makes this outcome measure symmetric around 0 and yields a corresponding sampling distribution that is closer to normality. Studies with zero events in one or both groups can be problematic, especially for the incidence rate ratio. Adding a small constant to the number of events is a common solution to this problem. When `to="only0"` (the default), the value of add (the default is 1/2) is added to x1i and x2i only in the studies that have zero events in one or both groups. When `to="all"`, the value of add is added to x1i and x2i in all studies. When `to="if0all"`, the value of add is added to x1i and x2i in all studies, but only when there is at least one study with zero events in one or both groups. Setting `to="none"` or add=0 has the same effect: No adjustment to the observed number of events is made. Depending on the outcome measure and the data, this may lead to division by zero inside of the function (when this occurs, the resulting value is recoded to NA). Like for $2 \times 2$ table data, studies where x1i=x2i=0 may be considered to be uninformative about the size of the effect and dropping such studies has sometimes been recommended. This can be done by setting drop00=TRUE. The values for such studies will then be set to NA.
A dataset corresponding to data of this type is provided in `dat.hart1999`.

*Measures for Quantitative Variables:*
When the response or dependent variable assessed in the individual studies is measured on some quantitative scale, it is customary to report certain summary statistics, such as the mean and standard deviation of the scores. The data layout for a study comparing two groups with respect to such a variable is then of the form:

|         | mean | standard deviation | group size |
|---------|------|--------------------|------------|
| group 1 | m1i  | sd1i               | n1i        |
| group 2 | m2i  | sd2i               | n2i        |

where m1i and m2i are the observed means of the two groups, sd1i and sd2i the observed standard deviations, and n1i and n2i the number of individuals in each group. Again, the two groups may be experimentally created (e.g., a treatment and control group based on random assignment) or naturally occurring (e.g., men and women). In either case, the raw mean difference, the standardized mean difference, and the ratio of means (also called response ratio) are useful outcome measures when meta-analyzing studies of this type (e.g., Borenstein, 2009).
The options for the `measure` argument are then:

- `"MD"` for the *raw mean difference*.
- `"SMD"` for the *standardized mean difference*.
- `"SMDH"` for the *standardized mean difference* with heteroscedastic population variances in the two groups (Bonett, 2008, 2009).
- `"ROM"` for the *log transformed ratio of means* (Hedges et al., 1999).

Note that the log is taken of the ratio of means, which makes this outcome measures symmetric around 0 and yields a corresponding sampling distribution that is closer to normality (however, note that if m1i and m2i have opposite signs, this outcome measure cannot be computed).
The negative bias in the standardized mean difference is automatically corrected for within the function, yielding Hedges' g for measure="SMD" (Hedges, 1981). Similarly, the same bias correction is applied for measure="SMDH" (Bonett, 2009). Finally, for measure="SMD", one can

choose between vtype="LS" (the default) and vtype="UB". The former uses a large sample approximation to compute the sampling variances. The latter provides unbiased estimates of the sampling variances.

A dataset corresponding to data of this type is provided in `dat.normand1999` (for mean differences and standardized mean differences). A dataset showing the use of the ratio of means measure is provided in `dat.curtis1998`.

It is also possible to transform standardized mean differences into log odds ratios (e.g., Cox & Snell, 1989; Chinn, 2000; Hasselblad & Hedges, 1995; Sanchez-Meca et al., 2003). The options for the measure argument are then:

- "D2ORN" for the *transformed standardized mean difference* as an estimate of the log odds ratio (normal distributions).
- "D2ORL" for the *transformed standardized mean difference* as an estimate of the log odds ratio (logistic distributions).

Both of these transformations provide an estimate of the log odds ratio, the first assuming that the responses within the two groups are normally distributed, while the second assumes that the responses follow logistic distributions.

A dataset illustrating the combined analysis of standardized mean differences and probit transformed risk differences is provided in `dat.gibson2002`.

### Outcome Measures for Variable Association:

Meta-analyses are often used to synthesize studies that examine the direction and strength of the association between two variables measured concurrently and/or without manipulation by experimenters. In this section, a variety of outcome measures will be discussed that may be suitable for a meta-analyses with this purpose. We can distinguish between measures that are applicable when both variables are measured on quantitative scales, when both variables measured are dichotomous, and when the two variables are of mixed types.

*Measures for Two Quantitative Variables:*

The (Pearson or product moment) correlation coefficient quantifies the direction and strength of the (linear) relationship between two quantitative variables and is therefore frequently used as the outcome measure for meta-analyses (e.g., Borenstein, 2009). Two alternative measures are a bias-corrected version of the correlation coefficient and Fisher's r-to-z transformed coefficient. For these measures, one needs to specify ri, the vector with the raw correlation coefficients, and ni, the corresponding sample sizes. The options for the measure argument are then:

- "COR" for the *raw correlation coefficient*.
- "UCOR" for the *raw correlation coefficient* corrected for its slight negative bias (based on equation 2.7 in Olkin & Pratt, 1958).
- "ZCOR" for the *Fisher's r-to-z transformed correlation coefficient* (Fisher, 1921).

For measure="COR" and measure="UCOR", one can choose between vtype="LS" (the default) and vtype="UB". The former uses a large sample approximation to compute the sampling variances. The latter provides approximately unbiased estimates of the sampling variances (see Hedges, 1989).

Datasets corresponding to data of this type are provided in `dat.mcdaniel1994` and `dat.molloy2014`.

*Measures for Two Dichotomous Variables:*

When the goal of a meta-analysis is to examine the relationship between two dichotomous variables, the data for each study can again be presented in the form of a $2 \times 2$ table, except that there may not be a clear distinction between the group (i.e., the row) and the outcome (i.e., the column) variable. Moreover, the table may be a result of cross-sectional (i.e., multinomial)

sampling, where none of the table margins (except the total sample size) is fixed by the study design.

The phi coefficient and the odds ratio are commonly used measures of association for $2 \times 2$ table data (e.g., Fleiss & Berlin, 2009). The latter is particularly advantageous, as it is directly comparable to values obtained from stratified sampling (as described earlier). Yule's Q and Yule's Y (Yule, 1912) are additional measures of association for $2 \times 2$ table data (although they are not typically used in meta-analyses). Finally, assuming that the two dichotomous variables are actually dichotomized versions of the responses on two underlying quantitative scales (and assuming that the two variables follow a bivariate normal distribution), it is also possible to estimate the correlation between the two variables using the tetrachoric correlation coefficient (Pearson, 1900; Kirk, 1973).

For any of these outcome measures, one needs to specify the cell frequencies via the `ai`, `bi`, `ci`, and `di` arguments. The options for the `measure` argument are then:

- `"OR"` for the *log odds ratio*.
- `"PHI"` for the *phi coefficient*.
- `"YUQ"` for *Yule's Q* (Yule, 1912).
- `"YUY"` for *Yule's Y* (Yule, 1912).
- `"RTET"` for the *tetrachoric correlation*.

Tables with one or two zero counts are handled as described earlier.

*Measures for Mixed Variable Types:*

Finally, we will consider outcome measures that can be used to describe the relationship between two variables, where one variable is dichotomous and the other variable measures some quantitative characteristic. In that case, it is likely that study authors again report summary statistics, such as the mean and standard deviation of the scores within the two groups (defined by the dichotomous variable). In that case, one can compute the point-biserial (Tate, 1954) as a measure of association between the two variables. If the dichotomous variable is actually a dichotomized version of the responses on an underlying quantitative scale (and assuming that the two variables follow a bivariate normal distribution), it is also possible to estimate the correlation between the two variables using the biserial correlation coefficient (Pearson, 1909; Soper, 1914).

Here, one again needs to specify `m1i` and `m2i` for the observed means of the two groups, `sd1i` and `sd2i` for the observed standard deviations, and `n1i` and `n2i` for the number of individuals in each group. The options for the `measure` argument are then:

- `"RPB"` for the *point-biserial correlation*.
- `"RBIS"` for the *biserial correlation*.

For `measure="RPB"`, one must indicate via `vtype="ST"` or `vtype="CS"` whether the data for the studies were obtained using stratified or cross-sectional (i.e., multinomial) sampling, respectively (it is also possible to specify an entire vector for the `vtype` argument in case the sampling schemes differed for the various studies).

## Outcome Measures for Individual Groups:

In this section, outcome measures will be described which may be useful when the goal of a meta-analysis is to synthesize studies that characterize some property of individual groups. We will again distinguish between measures that are applicable when the characteristic of interest is a dichotomous variable, when the characteristic represents an event count, or when the characteristic assessed is a quantiative variable.

*Measures for Dichotomous Variables:*
A meta-analysis may be conducted to aggregate studies that provide data for individual groups with respect to a dichotomous dependent variable. Here, one needs to specify xi and ni, denoting the number of individuals experiencing the event of interest and the total number of individuals, respectively. Instead of specifying ni, one can use mi to specify the number of individuals that do not experience the event of interest. The options for the measure argument are then:

- "PR" for the *raw proportion*.
- "PLN" for the *log transformed proportion*.
- "PLO" for the *logit transformed proportion* (i.e., log odds).
- "PAS" for the *arcsine square-root transformed proportion* (i.e., the angular transformation).
- "PFT" for the *Freeman-Tukey double arcsine transformed proportion* (Freeman & Tukey, 1950).

Zero cell entries can be problematic for certain outcome measures. When to="only0" (the default), the value of add (the default is 1/2) is added to xi and mi only for studies where xi or mi is equal to 0. When to="all", the value of add is added to xi and mi in all studies. When to="if0all", the value of add is added in all studies, but only when there is at least one study with a zero value for xi or mi. Setting to="none" or add=0 has the same effect: No adjustment to the observed values is made. Depending on the outcome measure and the data, this may lead to division by zero inside of the function (when this occurs, the resulting value is recoded to NA). Datasets corresponding to data of this type are provided in dat.pritz1997 and dat.debruin2009.

*Measures for Event Counts:*
Various measures can be used to characterize individual groups when the dependent variable assessed is an event count. Here, one needs to specify xi and ti, denoting the total number of events that occurred and the total person-times at risk, respectively. The options for the measure argument are then:

- "IR" for the *raw incidence rate*.
- "IRLN" for the *log transformed incidence rate*.
- "IRS" for the *square-root transformed incidence rate*.
- "IRFT" for the *Freeman-Tukey transformed incidence rate* (Freeman & Tukey, 1950).

Studies with zero events can be problematic, especially for the log transformed incidence rate. Adding a small constant to the number of events is a common solution to this problem. When to="only0" (the default), the value of add (the default is 1/2) is added to xi only in the studies that have zero events. When to="all", the value of add is added to xi in all studies. When to="if0all", the value of add is added to xi in all studies, but only when there is at least one study with zero events. Setting to="none" or add=0 has the same effect: No adjustment to the observed number of events is made. Depending on the outcome measure and the data, this may lead to division by zero inside of the function (when this occurs, the resulting value is recoded to NA).

*Measures for Quantitative Variables:*
The goal of a meta-analysis may also be to characterize individual groups, where the response, characteristic, or dependent variable assessed in the individual studies is measured on some quantitative scale. In the simplest case, the raw mean for the quantitative variable is reported for each group, which then becomes the observed outcome for the meta-analysis. Here, one needs to specify mi, sdi, and ni for the observed means, the observed standard deviations, and the sample sizes, respectively. The only option for the measure argument is then:

- "MN" for the *raw mean*.

Note that sdi is used to specify the standard deviations of the observed values of the response, characteristic, or dependent variable and not the standard errors of the means.

A more complicated situation arises when the purpose of the meta-analysis is to assess the amount of change within individual groups. In that case, either the raw mean change or standardized versions thereof can be used as outcome measures (Becker, 1988; Gibbons et al., 1993; Morris, 2000). Here, one needs to specify m1i and m2i, the observed means at the two measurement occasions, sd1i and sd2i for the corresponding observed standard deviations, ri for the correlation between the scores observed at the two measurement occasions, and ni for the sample size. The options for the measure argument are then:

- "MC" for the *raw mean change*.
- "SMCC" for the *standardized mean change* using change score standardization.
- "SMCR" for the *standardized mean change* using raw score standardization.

See also Morris and DeShon (2002) for a thorough discussion of the difference between the change score measures.

A few notes about the change score measures. In practice, one often has a mix of information available from the individual studies to compute these measures. In particular, if m1i and m2i are unknown, but the raw mean change is directly reported in a particular study, then you can set m1i to that value and m2i to 0 (making sure that the raw mean change was computed as m1i-m2i within that study and not the other way around). Also, for the raw mean change ("MC") or the standardized mean change using change score standardization ("SMCC"), if sd1i, sd2i, and ri are unknown, but the standard deviation of the change scores is directly reported, then you can set sd1i to that value and both sd2i and ri to 0. Finally, for the standardized mean change using raw score standardization ("SMCR"), argument sd2i is actually not needed, as the standardization is only based on sd1i (Becker, 1988; Morris, 2000), which is usually the pre-test standard deviation (if the post-test standard deviation should be used, then set sd1i to that).

**Other Outcome Measures for Meta-Analyses:**

Other outcome measures are sometimes used for meta-analyses that do not directly fall into the categories above. These are described in this section.

*Cronbach's alpha and Transformations Thereof:*

Meta-analytic methods can also be used to aggregate Cronbach's alpha values. This is usually referred to as a 'reliability generalization meta-analysis' (Vacha-Haase, 1998). Here, one needs to specify ai, mi, and ni for the observed alpha values, the number of items/replications/parts of the measurement instrument, and the sample sizes, respectively. One can either directly analyze the raw Cronbach's alpha values or transformations thereof (Bonett, 2002, 2010; Hakstian & Whalen, 1976). The options for the measure argument are then:

- "ARAW" for *raw alpha* values.
- "AHW" for *transformed alpha values* (Hakstian & Whalen, 1976).
- "ABT" for *transformed alpha values* (Bonett, 2002).

Note that the transformations implemented here are slightly different from the ones described by Hakstian and Whalen (1976) and Bonett (2002). In particular, for "AHW", the transformation $1 - (1 - \alpha)^{1/3}$ is used, while for "ABT", the transformation $-ln(1 - \alpha)$ is used. This ensures that the transformed values are monotonically increasing functions of $\alpha$.

A dataset corresponding to data of this type is provided in dat.bonett2010.

**Formula Interface:**

There are two general ways of specifying the data for computing the various effect size or outcome measures when using the `escalc` function, the default and a formula interface. When using the default interface, which is described above, the information needed to compute the various outcome measures is passed to the function via the various arguments outlined above (i.e., arguments `ai` through `ni`).

The formula interface works as follows. As above, the argument `measure` is a character string specifying which outcome measure should be calculated. The `formula` argument is then used to specify the data structure as a multipart formula. The `data` argument can be used to specify a data frame containing the variables in the formula. The `add`, `to`, and `vtype` arguments work as described above.

*Outcome Measures for Two-Group Comparisons:*

*Measures for Dichotomous Variables:*
For $2 \times 2$ table data, the `formula` argument takes the form `outcome ~ group | study`, where `group` is a two-level factor specifying the rows of the tables, `outcome` is a two-level factor specifying the columns of the tables (the two possible outcomes), and `study` is a factor specifying the study factor. The `weights` argument is used to specify the frequencies in the various cells.

*Measures for Event Counts:*
For two-group comparisons with event counts, the `formula` argument takes the form `events/times ~ group | study`, where `group` is a two-level factor specifying the group factor and `study` is a factor specifying the study factor. The left-hand side of the formula is composed of two parts, with the first variable for the number of events and the second variable for the person-times at risk.

*Measures for Quantitative Variables:*
For two-group comparisons with quantitative variables, the `formula` argument takes the form `means/sds ~ group | study`, where `group` is a two-level factor specifying the group factor and `study` is a factor specifying the study factor. The left-hand side of the formula is composed of two parts, with the first variable for the means and the second variable for the standard deviations. The `weights` argument is used to specify the sample sizes in the groups.

*Outcome Measures for Variable Association:*

*Measures for Two Quantitative Variables:*
For these outcome measures, the `formula` argument takes the form `outcome ~ 1 | study`, where `outcome` is used to specify the observed correlations and `study` is a factor specifying the study factor. The `weights` argument is used to specify the sample sizes.

*Measures for Two Dichotomous Variables:*
Here, the data layout is assumed to be the same as for two-group comparisons with dichotomous variables. Hence, the `formula` argument is specified in the same manner.

*Measures for Mixed Variable Types:*
Here, the data layout is assumed to be the same as for two-group comparisons with quantitative variables. Hence, the `formula` argument is specified in the same manner.

*Outcome Measures for Individual Groups:*

*Measures for Dichotomous Variables:*
For these outcome measures, the `formula` argument takes the form `outcome ~ 1 | study`, where `outcome` is a two-level factor specifying the columns of the tables (the two possible outcomes) and `study` is a factor specifying the study factor. The `weights` argument is used to specify the frequencies in the various cells.

*Measures for Event Counts:*

For these outcome measures, the `formula` argument takes the form `events/times ~ 1 | study`, where `study` is a factor specifying the study factor. The left-hand side of the formula is composed of two parts, with the first variable for the number of events and the second variable for the person-times at risk.

*Measures for Quantitative Variables:*

For this outcome measures, the `formula` argument takes the form `means/sds ~ 1 | study`, where `study` is a factor specifying the study factor. The left-hand side of the formula is composed of two parts, with the first variable for the means and the second variable for the standard deviations. The `weights` argument is used to specify the sample sizes.

Note: The formula interface is (currently) not implemented for the raw mean change and the standardized mean change measures.

*Other Outcome Measures for Meta-Analyses:*

*Cronbach's alpha and Transformations Thereof:*

For these outcome measures, the `formula` argument takes the form `alpha/items ~ 1 | study`, where `study` is a factor specifying the study factor. The left-hand side of the formula is composed of two parts, with the first variable for the Cronbach's alpha values and the second variable for the number of items.

## Value

An object of class `c("escalc","data.frame")`. The object is a data frame containing the following components:

| | |
|---|---|
| `yi` | observed outcomes or effect size estimates. |
| `vi` | corresponding (estimated) sampling variances. |

If `append=TRUE` and a data frame was specified via the `data` argument, then `yi` and `vi` are append to this data frame. Note that the `var.names` argument actually specifies the names of these two variables.

If the data frame already contains two variables with names as specified by the `var.names` argument, the values for these two variables will be overwritten when `replace=TRUE` (which is the default). By setting `replace=FALSE`, only values that are `NA` will be replaced.

The object is formated and printed with the [`print.escalc`](#) function. The [`summary.escalc`](#) function can be used to obtain confidence intervals for the individual outcomes.

## Note

The variable names specified under `var.names` should be syntactically valid variable names. If necessary, they are adjusted so that they are.

For standard meta-analyses using the typical (wide-format) data layout (i.e., one row in the dataset per study), the default interface is typically easier to use. The advantage of the formula interface is that it can, in principle, handle more complicated data structures (e.g., studies with more than two treatment groups or more than two outcomes). While such functionality is currently not implemented, this may be the case in the future.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

**References**

Bagos, P. G., & Nikolopoulos, G. K. (2009). Mixed-effects Poisson regression models for meta-analysis of follow-up studies with constant or varying durations. *The International Journal of Biostatistics*, **5**(1), article 21.

Becker, B. J. (1988). Synthesizing standardized mean-change measures. *British Journal of Mathematical and Statistical Psychology*, **41**, 257–278.

Bonett, D. G. (2002). Sample size requirements for testing and estimating coefficient alpha. *Journal of Educational and Behavioral Statistics*, **27**, 335–340.

Bonett, D. G. (2008). Confidence intervals for standardized linear contrasts of means. *Psychological Methods*, **13**, 99–109.

Bonett, D. G. (2009). Meta-analytic interval estimation for standardized and unstandardized mean differences. *Psychological Methods*, **14**, 225–238.

Bonett, D. G. (2010). Varying coefficient meta-analytic methods for alpha reliability. *Psychological Methods*, **15**, 368–385.

Borenstein, M. (2009). Effect sizes for continuous data. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 221–235). New York: Russell Sage Foundation.

Chinn, S. (2000). A simple method for converting an odds ratio to effect size for use in meta-analysis. *Statistics in Medicine*, **19**, 3127–3131.

Cox, D. R., & Snell, E. J. (1989). *Analysis of binary data* (2nd ed.). London: Chapman & Hall.

Fisher, R. A. (1921). On the "probable error" of a coefficient of correlation deduced from a small sample. *Metron*, **1**, 1–32.

Fleiss, J. L., & Berlin, J. (2009). Effect sizes for dichotomous data. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 237–253). New York: Russell Sage Foundation.

Freeman, M. F., & Tukey, J. W. (1950). Transformations related to the angular and the square root. *Annals of Mathematical Statistics*, **21**, 607–611.

Gibbons, R. D., Hedeker, D. R., & Davis, J. M. (1993). Estimation of effect size from a series of experiments involving paired comparisons. *Journal of Educational Statistics*, **18**, 271–279.

Hakstian, A. R., & Whalen, T. E. (1976). A k-sample significance test for independent alpha coefficients. *Psychometrika*, **41**, 219–231.

Hasselblad, V., & Hedges, L. V. (1995). Meta-analysis of screening and diagnostic tests. Psychological Bulletin, 117(1), 167-178.

Hedges, L. V. (1981). Distribution theory for Glass's estimator of effect size and related estimators. *Journal of Educational Statistics*, **6**, 107–128.

Hedges, L. V. (1989). An unbiased correction for sampling error in validity generalization studies. *Journal of Applied Psychology*, **74**, 469–477.

Hedges, L. V., Gurevitch, J., & Curtis, P. S. (1999). The meta-analysis of response ratios in experimental ecology. *Ecology*, **80**, 1150–1156.

Higgins, J. P. T., & Green, S. (Eds.) (2008). *Cochrane handbook for systematic reviews of interventions*. Chichester, Englang: Wiley.

Kirk, D. B. (1973). On the numerical approximation of the bivariate normal (tetrachoric) correlation coefficient. *Psychometrika*, **38**, 259–268.

Morris, S. B. (2000). Distribution of the standardized mean change effect size for meta-analysis on repeated measures. *British Journal of Mathematical and Statistical Psychology*, **53**, 17–29.

Morris, S. B., & DeShon, R. P. (2002). Combining effect size estimates in meta-analysis with repeated measures and independent-groups designs. *Psychological Methods*, **7**, 105–125.

Rücker, G., Schwarzer, G., Carpenter, J., & Olkin, I. (2009). Why add anything to nothing? The arcsine difference as a measure of treatment effect in meta-analysis with zero cells. *Statistics in Medicine*, **28**, 721–738.

Sánchez-Meca, J., Marín-Martínez, F., & Chacón-Moscoso, S. (2003). Effect-size indices for dichotomized outcomes in meta-analysis. *Psychological Methods*, **8**, 448–467.

Olkin, I., & Pratt, J. W. (1958). Unbiased estimation of certain correlation coefficients. *Annals of Mathematical Statistics*, **29**, 201–211.

Pearson, K. (1900). Mathematical contribution to the theory of evolution. VII. On the correlation of characters not quantitatively measurable. *Philosophical Transactions of the Royal Society of London, Series A*, **195**, 1–47.

Pearson, K. (1909). On a new method of determining correlation between a measured character A, and a character B, of which only the percentage of cases wherein B exceeds (or falls short of) a given intensity is recorded for each grade of A. *Biometrika*, **7**, 96–105.

Rothman, K. J., Greenland, S., & Lash, T. L. (2008). *Modern epidemiology* (3rd ed.). Philadelphia: Lippincott Williams & Wilkins.

Soper, H. E. (1914). On the probable error of the bi-serial expression for the correlation coefficient. *Biometrika*, **10**, 384–390.

Tate, R. F. (1954). Correlation between a discrete and a continuous variable: Point-biserial correlation. *Annals of Mathematical Statistics*, **25**, 603–607.

Vacha-Haase, T. (1998). Reliability generalization: Exploring variance in measurement error affecting score reliability across studies. *Educational and Psychological Measurement*, **58**, 6–20.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

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

Yusuf, S., Peto, R., Lewis, J., Collins, R., & Sleight, P. (1985). Beta blockade during and after myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Disease*, **27**, 335–371.

## See Also

print.escalc, summary.escalc, rma.uni, rma.mh, rma.peto, rma.glmm

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
dat

### suppose that for a particular study, yi and vi are known (i.e., have
### already been calculated) but the 2x2 table counts are not known; with
### replace=FALSE, the yi and vi values for that study are not replaced
dat[1:12,10:11] <- NA
dat[13,4:7] <- NA
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat, replace=FALSE)
dat

### using formula interface (first rearrange data into long format)
dat.long <- to.long(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
                    data=dat.bcg, append=FALSE, vlong=TRUE)
escalc(measure="RR", outcome ~ group | study, weights=freq, data=dat.long)
```

---

| fitstats | *Fit Statistics and Information Criteria for 'rma' Objects* |
|---|---|

---

## Description

Functions to extract the log-likelihood, deviance, AIC, BIC, and AICc values from objects of class
"rma".

## Usage

```
fitstats(object, ...)

## S3 method for class 'rma'
fitstats(object, REML, ...)

## S3 method for class 'rma'
logLik(object, REML, ...)
## S3 method for class 'rma'
deviance(object, REML, ...)

## S3 method for class 'rma'
AIC(object, ..., k=2, correct=FALSE)
## S3 method for class 'rma'
BIC(object, ...)
```

## Arguments

| | |
|---|---|
| object | an object of class "rma". |
| REML | logical indicating whether the regular or restricted likelihood function should be used to obtain the fit statistics and information criteria. Defaults to the method of estimation used, that is TRUE if object was fitted with method="REML" and FALSE otherwise. |
| k | numeric value specifying the penalty per parameter to be used. The default (k=2) is the classical AIC. See [AIC](#) for more details. |
| correct | logical indicating whether the regular (default) or corrected (i.e., AICc) should be extracted. |
| ... | other arguments. |

## Value

For fitstats.rma, a column vector with the (restricted) log-likelihood, deviance, AIC, BIC, and AICc values.

For logLik.rma, an object of class "logLik", providing the (restricted) log-likelihood of the model evaluated at the estimated coefficient(s).

For deviance.rma, a numeric value with the corresponding deviance.

For AIC.rma, a numeric value with the corresponding AIC or AICc.

For BIC.rma, a numeric value with the corresponding BIC.

## Note

Note that variance components in the model (e.g., $\tau^2$ in random/mixed-effects models) are counted as additional parameters in the calculation of the AIC, BIC, and AICc. Also, the fixed effects are counted as parameters in the calculation of the AIC, BIC, and AICc even when using REML estimation.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[rma.uni](#), [rma.mh](#), [rma.peto](#), [rma.glmm](#), [rma.mv](#), [anova.rma](#)

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res1 <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
            data=dat.bcg, method="ML")

### mixed-effects model with two moderators (latitude and publication year)
res2 <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat + year,
            data=dat.bcg, method="ML")

fitstats(res1)
fitstats(res2)

logLik(res1)
logLik(res2)

deviance(res1)
deviance(res2)

AIC(res1)
AIC(res2)

AIC(res1, correct=TRUE)
AIC(res2, correct=TRUE)

BIC(res1)
BIC(res2)
```

---

fitted.rma                *Fitted Values for 'rma' Objects*

---

## Description

The function computes the fitted values for objects of class "rma".

## Usage

```
## S3 method for class 'rma'
fitted(object, ...)
```

## Arguments

object          an object of class "rma".
...             other arguments.

## Value

A vector with the fitted values.

## Note

The `predict.rma` function also provides standard errors and confidence intervals for the fitted values. Best linear unbiased predictions (BLUPs) that combine the fitted values based on the fixed effects and the estimated contributions of the random effects can be obtained with `blup.rma.uni` (only for objects of class "rma.uni").

For objects not involving moderators, the fitted values are all identical to the estimated value of the model intercept.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

predict.rma, blup.rma.uni

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
fitted(res)
```

---

forest *Forest Plots*

---

## Description

The function `forest` is generic. It can be used to create forest plots.

## Usage

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

## Arguments

x                       either an object of class `"rma"`, a vector with the observed effect size or out-
                        comes, or an object of class `"cumul.rma"`. See 'Details'.

...                     other arguments.

## Details

Currently, methods exist for three types of situations.

In the first case, object x is a fitted model object coming from the `rma.uni`, `rma.mh`, or `rma.peto`
functions. The corresponding method is then `forest.rma`.

Alternatively, object x can be a vector with observed effect size or outcomes. The corresponding
method is then `forest.default`.

Finally, object x could be an object coming from the `cumul.rma.uni`, `cumul.rma.mh`, or `cumul.rma.peto`
functions. The corresponding method is then `forest.cumul.rma`.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Lewis, S., & Clarke, M. (2001). Forest plots: Trying to see the wood and the trees. *British Medical
Journal*, **322**, 1479–1480.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of
Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

forest.rma, forest.default, forest.cumul.rma

---

forest.cumul.rma              *Forest Plots (Method for 'cumul.rma' Objects)*

---

## Description

Function to create forest plots for objects of class `"cumul.rma"`.

## Usage

```
## S3 method for class 'cumul.rma'
forest(x, annotate=TRUE,
       xlim, alim, clim, ylim, at, steps=5,
       level=x$level, digits=2, refline=0, xlab,
       ilab, ilab.xpos, ilab.pos,
       transf, atransf, targs, rows,
       efac=1, pch=15, psize=1,
       cex, cex.lab, cex.axis, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "cumul.rma". |
| annotate | logical specifying whether annotations should be added to the plot (default is TRUE). |
| xlim | horizontal limits of the plot region. If unspecified, the function tries to set the horizontal plot limits to some sensible values. |
| alim | the actual x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| clim | limits for the confidence intervals. If unspecified, no limits are used. |
| ylim | the y limits of the plot. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| at | position of the x-axis tick marks and corresponding labels. If unspecified, the function tries to set the tick mark positions/labels to some sensible values. |
| steps | the number of tick marks for the x-axis (default is 5). Ignored when the user specifies the positions via the at argument. |
| level | numerical value between 0 and 100 specifying the confidence interval level (the default is to take the value from the object). |
| digits | integer specifying the number of decimal places to which the tick mark labels of the x-axis and the annotations should be rounded (default is 2). Can also be a vector of two integers, the first specifying the number of decimal places for the annotations, the second for the x-axis labels. |
| refline | value at which a vertical 'reference' line should be drawn (default is 0). The line can be suppressed by setting this argument to NA. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| ilab | optional vector or matrix of character strings providing additional information about the studies. |
| ilab.xpos | vector of numerical value(s) specifying the x-axis position(s) of the character vector(s) given via ilab (must be specified if ilab is specified). |
| ilab.pos | integer(s) (either 1, 2, 3, or 4) specifying the alignment of the character vector(s) given via ilab (2 means right, 4 means left aligned). If unspecified, the default is to center the labels. |

| transf | optional argument specifying the name of a function that should be used to transform the observed effect sizes, summary estimates, fitted values, and confidence interval bounds (e.g., transf=exp). If unspecified, no transformation is used. |
| atransf | optional argument specifying the name of a function that should be used to transform the x-axis labels and annotations (e.g., transf=exp). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified via transf or atransf. |
| rows | optional vector specifying the rows (or more generally, the horizontal positions) for plotting the outcomes. If unspecified, the function sets this value automatically. Can also be a single value specifying the row (horizontal position) of the first outcome (the remaining outcomes are then plotted below this starting row). |
| efac | vertical expansion factor for arrows, confidence interval limits, and the symbol used to denote summary estimates. The default value of 1 should usually work okay. |
| pch | plotting symbol to use for the observed effect sizes or outcomes. By default, a filled square is used. See [points](points) for other options. Can also be a vector of values. |
| psize | optional vector with point sizes for the observed effect sizes or outcomes. Default is 1. |
| cex | optional character and symbol expansion factor. If unspecified, the function tries to set this to a sensible value. |
| cex.lab | optional expansion factor for the x-axis titel. If unspecified, the function tries to set this to a sensible value. |
| cex.axis | optional expansion factor for the x-axis labels. If unspecified, the function tries to set this to a sensible value. |
| ... | other arguments. |

### Details

The plot shows the estimated (average) outcome with corresponding confidence interval as one study at a time is added to the analysis.

### Note

The function tries to set some sensible values for the optional arguments, but it may be necessary to tweak these in certain circumstances.

The function actually returns some information about the chosen defaults invisibly. Printing this information is useful as a starting point to make adjustments to the plot.

If the number of studies is quite large, the labels, annotations, and symbols may become quite small and impossible to read. Stretching the plot window vertically may then provide a more readable figure (one should call the function again after adjusting the window size, so that the label/symbol sizes can be properly adjusted). Also, the cex, cex.lab, and cex.axis arguments are then useful to adjust the symbol and text sizes.

If the horizontal plot and/or x-axis limits are set by the user, then the horizontal plot limits (xlim) must be at least as wide as the x-axis limits (alim). Moreover, the x-axis limits must encompass the observed effect sizes or outcomes. These restrictions are enforced inside the function.

If the outcome measure used for creating the plot has absolute limits (e.g., correlations are bounded between -1 and 1, proportions are bounded between 0 and 1), one can use the `clim` argument to enforce those limits (confidence intervals cannot exceed those bounds then).

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Chalmers, T. C., & Lau, J. (1993). Meta-analytic stimulus for changes in clinical trials. *Statistical Methods in Medical Research*, **2**, 161–172.

Lau, J., Schmid, C. H., & Chalmers, T. C. (1995). Cumulative meta-analysis of clinical trials builds evidence for exemplary medical care. *Journal of Clinical Epidemiology*, **48**, 45–57.

Lewis, S., & Clarke, M. (2001). Forest plots: Trying to see the wood and the trees. *British Medical Journal*, **322**, 1479–1480.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

### See Also

forest, cumul

### Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res <- rma(yi, vi, data=dat, slab=paste(author, year, sep=", "))

x <- cumul(res, order=order(dat$year))
forest(x)
forest(x, alim=c(-2,1))

### meta-analysis of the (log) relative risks using the Mantel-Haenszel method
res <- rma.mh(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
              slab=paste(author, year, sep=", "))
x <- cumul(res, order=order(dat$year))
forest(x, alim=c(-2,1))
```

---

forest.default                    *Forest Plots (Default Method)*

---

**Description**

Function to create forest plots for a given set of data.

**Usage**

```
## Default S3 method:
forest(x, vi, sei, ci.lb, ci.ub, annotate=TRUE, showweights=FALSE,
       xlim, alim, clim, ylim, at, steps=5,
       level=95, digits=2, refline=0, xlab,
       slab, ilab, ilab.xpos, ilab.pos,
       subset, transf, atransf, targs, rows,
       efac=1, pch=15, psize, cex, cex.lab, cex.axis, ...)
```

**Arguments**

| | |
|---|---|
| x | vector of length $k$ with the observed effect sizes or outcomes. |
| vi | vector of length $k$ with the corresponding sampling variances. |
| sei | vector of length $k$ with the corresponding standard errors.<br>(note: only one of the two, vi or sei, needs to be specified) |
| ci.lb | vector of length $k$ with the corresponding lower confidence intervals bounds. Not needed if vi or sei is specified. See 'Details'. |
| ci.ub | vector of length $k$ with the corresponding upper confidence intervals bounds. Not needed if vi or sei is specified. See 'Details'. |
| annotate | logical specifying whether annotations should be added to the plot (default is TRUE). |
| showweights | logical specifying whether the annotations should also include inverse variance weights (default is FALSE). |
| xlim | horizontal limits of the plot region. If unspecified, the function tries to set the horizontal plot limits to some sensible values. |
| alim | the actual x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| clim | limits for the confidence intervals. If unspecified, no limits are used. |
| ylim | the y limits of the plot. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| at | position of the x-axis tick marks and corresponding labels. If unspecified, the function tries to set the tick mark positions/labels to some sensible values. |
| steps | the number of tick marks for the x-axis (default is 5). Ignored when the user specifies the positions via the at argument. |

| | |
|---|---|
| level | numerical value between 0 and 100 specifying the confidence interval level (default is 95). |
| digits | integer specifying the number of decimal places to which the tick mark labels of the x-axis and the annotations should be rounded (default is 2). Can also be a vector of two integers, the first specifying the number of decimal places for the annotations, the second for the x-axis labels. |
| refline | value at which a vertical 'reference' line should be drawn (default is 0). The line can be suppressed by setting this argument to NA. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| slab | optional vector with labels for the $k$ studies. If unspecified, simple labels are created within the function. To suppress labels, set this argument to NA. |
| ilab | optional vector or matrix of character strings providing additional information about the studies. |
| ilab.xpos | numerical value(s) specifying the x-axis position(s) of the character vector(s) given via ilab (must be specified if ilab is specified). |
| ilab.pos | integer(s) (either 1, 2, 3, or 4) specifying the alignment of the character vector(s) given via ilab (2 means right, 4 mean left aligned). If unspecified, the default is to center the labels. |
| subset | optional vector indicating the subset of studies that should be used for the plot. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the observations to include. Note that this argument can also be used for reordering the studies. |
| transf | optional argument specifying the name of a function that should be used to transform the observed effect sizes or outcomes and corresponding confidence interval bounds (e.g., transf=exp). If unspecified, no transformation is used. |
| atransf | optional argument specifying the name of a function that should be used to transform the x-axis labels and annotations (e.g., transf=exp). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified via transf or atransf. |
| rows | optional vector specifying the rows (or more generally, the horizontal positions) for plotting the outcomes. If unspecified, the function sets this value automatically. Can also be a single value specifying the row (horizontal position) of the first outcome (the remaining outcomes are then plotted below this starting row). |
| efac | vertical expansion factor for arrows, confidence interval limits, and the symbol used to denote summary estimates. The default value of 1 should usually work okay. |
| pch | plotting symbol to use for the observed effect sizes or outcomes. By default, a filled square is used. See [points](#) for other options. Can also be a vector of values. |
| psize | optional vector with point sizes for the observed effect sizes or outcomes. If unspecified, the point sizes are drawn proportional to the precision of the estimates. |

| cex | optional character and symbol expansion factor. If unspecified, the function tries to set this to a sensible value. |
|---|---|
| cex.lab | optional expansion factor for the x-axis titel. If unspecified, the function tries to set this to a sensible value. |
| cex.axis | optional expansion factor for the x-axis labels. If unspecified, the function tries to set this to a sensible value. |
| ... | other arguments. |

## Details

The plot shows the individual observed effect sizes or outcomes with corresponding confidence intervals. To use the function, one should specify the observed effect sizes or outcomes (via the x argument) together with the corresponding sampling variances (via the vi argument) or with the corresponding standard errors (via the sei argument). Alternatively, one can specify the observed effect sizes or outcomes together with the corresponding confidence interval bounds (via the ci.lb and ci.ub arguments).

With the transf argument, the observed effect sizes or outcomes and corresponding confidence interval bounds can be transformed with some suitable function. For example, when plotting log odds ratios, then one could use transf=exp to obtain a forest plot showing the odds ratios. Alternatively, one can use the atransf argument to transform the x-axis labels and annotations (e.g., atransf=exp). The examples below illustrate the use of these arguments.

By default, the studies are ordered from top to bottom (i.e., the first study in the dataset will be placed in row $k$, the second study in row $k-1$, and so on, until the last study, which is placed in the first row). The studies can be reordered with the subset argument (by specifying a vector with indices with the desired order).

Summary estimates can also be added to the plot with the [addpoly](addpoly) function. See the documentation for that function for examples.

## Note

The function tries to set some sensible values for the optional arguments, but it may be necessary to tweak these in certain circumstances.

The function actually returns some information about the chosen defaults invisibly. Printing this information is useful as a starting point to make adjustments to the plot.

If the number of studies is quite large, the labels, annotations, and symbols may become quite small and impossible to read. Stretching the plot window vertically may then provide a more readable figure (one should call the function again after adjusting the window size, so that the label/symbol sizes can be properly adjusted). Also, the cex, cex.lab, and cex.axis arguments are then useful to adjust the symbol and text sizes.

If the horizontal plot and/or x-axis limits are set by the user, then the horizontal plot limits (xlim) must be at least as wide as the x-axis limits (alim). Moreover, the x-axis limits must encompass the observed effect sizes or outcomes. These restrictions are enforced inside the function.

If the outcome measure used for creating the plot has absolute limits (e.g., correlations are bounded between -1 and 1, proportions are bounded between 0 and 1), one can use the clim argument to enforce those limits (confidence intervals cannot exceed those bounds then).

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Lewis, S., & Clarke, M. (2001). Forest plots: Trying to see the wood and the trees. *British Medical Journal*, **322**, 1479–1480.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[forest](), [forest.rma](), [addpoly]()

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### default forest plot of the observed log relative risks
forest(dat$yi, dat$vi)

### forest plot of the observed relative risks
forest(dat$yi, dat$vi, slab=paste(dat$author, dat$year, sep=", "), transf=exp,
       alim=c(0,2), steps=5, xlim=c(-2,3.5), refline=1)

### forest plot of the observed relative risks
forest(dat$yi, dat$vi, slab=paste(dat$author, dat$year, sep=", "), atransf=exp,
       at=log(c(.05,.25,1,4,20)), xlim=c(-10,8))

### see also examples for the forest.rma function
```

---

forest.rma *Forest Plots (Method for 'rma' Objects)*

---

## Description

Function to create forest plots for objects of class "rma".

## Usage

```
## S3 method for class 'rma'
forest(x, annotate=TRUE, addfit=TRUE, addcred=FALSE, showweights=FALSE,
       xlim, alim, clim, ylim, at, steps=5,
       level=x$level, digits=2, refline=0, xlab,
       slab, mlab, ilab, ilab.xpos, ilab.pos,
       order, transf, atransf, targs, rows,
       efac=1, pch=15, psize, col, border, lty,
       cex, cex.lab, cex.axis, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "rma". |
| annotate | logical specifying whether annotations should be added to the plot (default is TRUE). |
| addfit | logical specifying whether the summary estimate (for models without moderators) or fitted values (for models with moderators) should be added to the plot. See 'Details'. |
| addcred | logical specifying whether the bounds of the credibility/prediction interval should be added to the plot. See 'Details'. |
| showweights | logical specifying whether the annotations should also include the weights given to the observed effects or outcomes during the model fitting (default is FALSE). |
| xlim | horizontal limits of the plot region. If unspecified, the function tries to set the horizontal plot limits to some sensible values. |
| alim | the actual x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| clim | limits for the confidence/credibility/prediction intervals. If unspecified, no limits are used. |
| ylim | the y limits of the plot. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| at | position of the x-axis tick marks and corresponding labels. If unspecified, the function tries to set the tick mark positions/labels to some sensible values. |
| steps | the number of tick marks for the x-axis (default is 5). Ignored when the user specifies the positions via the at argument. |
| level | numerical value between 0 and 100 specifying the confidence interval level (the default is to take the value from the object). |
| digits | integer specifying the number of decimal places to which the tick mark labels of the x-axis and the annotations should be rounded (default is 2). Can also be a vector of two integers, the first specifying the number of decimal places for the annotations, the second for the x-axis labels. |
| refline | value at which a vertical 'reference' line should be drawn (default is 0). The line can be suppressed by setting this argument to NA. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |

| | |
|---|---|
| slab | optional vector with labels for the $k$ studies. If unspecified, the labels are either taken from the object (if study labels were specified) or simple labels are created within the function. To suppress labels, set this argument to NA. |
| mlab | optional character string giving a label to the summary estimate from a fixed- or random-effects model. If unspecified, the label is created within the function. |
| ilab | optional vector or matrix of character strings providing additional information about the studies. |
| ilab.xpos | vector of numerical value(s) specifying the x-axis position(s) of the character vector(s) given via ilab (must be specified if ilab is specified). |
| ilab.pos | integer(s) (either 1, 2, 3, or 4) specifying the alignment of the character vector(s) given via ilab (2 means right, 4 mean left aligned). If unspecified, the default is to center the labels. |
| order | optional character string specifying how the studies should be ordered. See 'Details'. |
| transf | optional argument specifying the name of a function that should be used to transform the observed effect sizes, summary estimates, fitted values, and confidence interval bounds (e.g., transf=exp). If unspecified, no transformation is used. |
| atransf | optional argument specifying the name of a function that should be used to transform the x-axis labels and annotations (e.g., transf=exp). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified via transf or atransf. |
| rows | optional vector specifying the rows (or more generally, the horizontal positions) for plotting the outcomes. If unspecified, the function sets this value automatically. Can also be a single value specifying the row (horizontal position) of the first outcome (the remaining outcomes are then plotted below this starting row). |
| efac | vertical expansion factor for arrows, confidence interval limits, and the symbol used to denote summary estimates. The default value of 1 should usually work okay. |
| pch | plotting symbol to use for the observed effect sizes or outcomes. By default, a filled square is used. See [points](#) for other options. Can also be a vector of values. |
| psize | optional vector with point sizes for the observed effect sizes or outcomes. If unspecified, the point sizes are drawn proportional to the precision of the estimates. |
| cex | optional character and symbol expansion factor. If unspecified, the function tries to set this to a sensible value. |
| cex.lab | optional expansion factor for the x-axis titel. If unspecified, the function tries to set this to a sensible value. |
| cex.axis | optional expansion factor for the x-axis labels. If unspecified, the function tries to set this to a sensible value. |
| col | optional character string specifying the name of a color to use for the summary polygon or fitted values. If unspecified, the function sets a default color. |

| border | optional character string specifying the name of a color to use for the border of the summary polygon or fitted values. If unspecified, the function sets a default color. |
| --- | --- |
| lty | optional character string specifying the line type for the confidence intervals (if unspecified, the function sets this to "solid" by default). |
| ... | other arguments. |

## Details

The plot shows the individual observed effect sizes or outcomes with corresponding confidence intervals.

For fixed- and random-effects models (i.e., for models without moderators), a polygon is added to the bottom of the forest plot, showing the summary estimate based on the model (with the outer edges of the polygon indicating the confidence interval limits). The col and border arguments can be used to adjust the (border) color of the polygon.

For random-effects models and if addcred=TRUE, a dotted line indicates the (approximate) bounds of the credibility/prediction interval (the interval indicates where level % of the true effects are expected to fall) (Riley et al., 2011). For random-effects models of class "rma.mv" (see rma.mv) with multiple $\tau^2$ values, the addcred argument can be used to specify for which level of the inner factor the credibility/prediction intervals should be provided (since the credibility/prediction intervals differ depending on the $\tau^2$ value).

For models involving moderators, the fitted value for each study is added as a polygon to the plot. By default, the width of the polygons corresponds to the confidence interval limits for the fitted values. By setting addcred=TRUE, the width reflects the credibility/prediction interval limits. Again, the col and border arguments can be used to adjust the (border) color of the polygons. These polygons can be suppressed by setting addfit=FALSE.

With the transf argument, the observed effect sizes or outcomes, summary estimate, fitted values, confidence interval bounds, and credibility/prediction interval bounds can be transformed with some suitable function. For example, when plotting log odds ratios, one could use transf=exp to obtain a forest plot showing the odds ratios. Alternatively, one can use the atransf argument to transform the x-axis labels and annotations (e.g., atransf=exp). The examples below illustrate the use of these arguments.

By default, the studies are ordered from top to bottom (i.e., the first study in the dataset will be placed in row $k$, the second study in row $k - 1$, and so on, until the last study, which is placed in the first row). The studies can be reordered with the order argument:

- order="obs": the studies are ordered by the observed effect sizes,
- order="fit": the studies are ordered by the fitted values,
- order="prec": the studies are ordered by their sampling variances,
- order="resid": the studies are ordered by the size of their residuals,
- order="rstandard": the studies are ordered by the size of their standardized residuals,
- order="abs.resid": the studies are ordered by the size of their absolute residuals,
- order="abs.rstandard": the studies are ordered by the size of their absolute standardized residuals.

Alternatively, it is also possible to set order equal to a vector with indices specifying the desired order (see examples below).

Additional summary estimates can also be added to the plot with the [addpoly](#) function. See the documentation for that function for examples.

### Note

The function tries to set some sensible values for the optional arguments, but it may be necessary to tweak these in certain circumstances.

The function actually returns some information about the chosen defaults invisibly. Printing this information is useful as a starting point to make adjustments to the plot (see examples below).

If the number of studies is quite large, the labels, annotations, and symbols may become quite small and impossible to read. Stretching the plot window vertically may then provide a more readable figure (one should call the function again after adjusting the window size, so that the label/symbol sizes can be properly adjusted). Also, the cex, cex.lab, and cex.axis arguments are then useful to adjust the symbol and text sizes.

If the horizontal plot and/or x-axis limits are set by the user, then the horizontal plot limits (xlim) must be at least as wide as the x-axis limits (alim). Moreover, the x-axis limits must encompass the observed effect sizes or outcomes. These restrictions are enforced inside the function.

If the outcome measure used for creating the plot has absolute limits (e.g., correlations are bounded between −1 and 1, proportions are bounded between 0 and 1), one can use the clim argument to enforce those limits (confidence/credibility/prediction intervals cannot exceed those bounds then).

The models without moderators, the col argument can also be a vector of two elements, the first for specifying the color of the summary polygon, the second for specifying the color of the line for the credibility/prediction interval.

The lty can also be a vector of two elements, the first for specifying the line type for the individual CIs ("solid" by default), the second for the line type of the credibility/prediction interval ("dotted" by default).

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

### References

Lewis, S., & Clarke, M. (2001). Forest plots: Trying to see the wood and the trees. *British Medical Journal*, **322**, 1479–1480.

Riley, R. D., Higgins, J. P., & Deeks, J. J. (2011). Interpretation of random effects meta-analyses. *British Medical Journal*, **342**, d549.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

### See Also

[forest](#), [forest.default](#), [addpoly](#)

**Examples**

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
           slab=paste(author, year, sep=", "))

### default forest plot of the log relative risks and summary estimate
forest(res)

### summary estimate in row -1; studies in rows k=13 through 1; horizontal
### lines in rows 0 and k+1; and two extra lines of space at the top for
### headings or other annotations
op <- par(xpd=TRUE)
text(x=-8.5, y=-1:16, -1:16, pos=4, cex=.5)
par(op)

### can also inspect defaults chosen
defaults <- forest(res)
defaults

### several forest plots illustrating the use of various arguments
forest(res, cex=.8)
forest(res, cex=.8, addcred=TRUE)
forest(res, cex=.8, alim=c(-3,3))
forest(res, cex=.8, order="prec", alim=c(-3,3))
forest(res, cex=.8, order=order(dat.bcg$ablat), addcred=TRUE)

### adjust xlim values to see how that changes the plot
forest(res)
par("usr")[1:2] ### this shows what xlim values were chosen by default
forest(res, xlim=c(-16,14))
forest(res, xlim=c(-18,10))
forest(res, xlim=c(-10,10))

### illustrate transf argument
forest(res, transf=exp, at=c(0,1,2,4,6), xlim=c(-8,12), cex=.8, refline=1)

### illustrate atransf argument
forest(res, atransf=exp, at=log(c(.05,.25,1,4,20)), xlim=c(-8,7), cex=.8)

### showweights argument
forest(res, atransf=exp, at=log(c(.05,.25,1,4,20)), xlim=c(-8,8),
       order="prec", showweights=TRUE, cex=.8)

### forest plot with extra annotations
forest(res, atransf=exp, at=log(c(.05, .25, 1, 4)), xlim=c(-16,6),
       ilab=cbind(dat.bcg$tpos, dat.bcg$tneg, dat.bcg$cpos, dat.bcg$cneg),
       ilab.xpos=c(-9.5,-8,-6,-4.5), cex=.75)
op <- par(cex=.75, font=2)
text(c(-9.5,-8,-6,-4.5), 15, c("TB+", "TB-", "TB+", "TB-"))
```

```
text(c(-8.75,-5.25),      16, c("Vaccinated", "Control"))
text(-16,                 15, "Author(s) and Year",      pos=4)
text(6,                   15, "Relative Risk [95% CI]", pos=2)
par(op)

### mixed-effects model with absolute latitude in the model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat,
           data=dat.bcg, slab=paste(author, year, sep=", "))

### forest plot with observed and fitted values
forest(res, xlim=c(-9,5), order="fit", cex=.8, ilab=dat.bcg$ablat,
       ilab.xpos=-4, atransf=exp, at=log(c(.05,.25,1,4)))
op <- par(font=2)
text(-9, 15, "Author(s) and Year",    pos=4, cex=.8)
text( 5, 15, "Observed RR [95% CI]",  pos=2, cex=.8)
text(-4, 15, "Latitude",                     cex=.8)
par(op)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg,
           slab=paste(author, year, sep=", "))

### for more complicated plots, the ylim and rows arguments may be useful
forest(res)
forest(res, ylim=c(-1.5, 16)) ### the default
forest(res, ylim=c(-1.5, 20)) ### extra space in plot
forest(res, ylim=c(-1.5, 20), rows=c(17:15, 12:6, 3:1)) ### set positions

### forest plot with subgrouping of studies
forest(res, xlim=c(-16, 6), at=log(c(.05, .25, 1, 4)), atransf=exp,
       ilab=cbind(dat.bcg$tpos, dat.bcg$tneg, dat.bcg$cpos, dat.bcg$cneg),
       ilab.xpos=c(-9.5,-8,-6,-4.5), cex=.75, ylim=c(-1, 21),
       order=order(dat.bcg$alloc), rows=c(1:2,5:11,14:17))
op <- par(cex=.75, font=4)
text(-16, c(18,12,3), c("Systematic Allocation", "Random Allocation",
                        "Alternate Allocation"), pos=4)
par(cex=.75, font=2)
text(c(-9.5,-8,-6,-4.5), 20, c("TB+", "TB-", "TB+", "TB-"))
text(c(-8.75,-5.25),     21, c("Vaccinated", "Control"))
text(-16,                20, "Author(s) and Year",     pos=4)
text(6,                  20, "Relative Risk [95% CI]", pos=2)
par(op)

### see also the addpoly.rma function for an example where summaries
### for the three subgroups are added to such a forest plot

### illustrate use of clim argument with a meta-analysis of raw correlation
### coefficients (data from Pritz, 1997); without clim=c(0,1), some of the
### CIs have upper bounds larger than 1
dat <- escalc(measure="PR", xi=xi, ni=ni, data=dat.pritz1997)
res <- rma(yi, vi, data=dat, slab=paste0(study, ") ", authors))
forest(res, xlim=c(-1,2), alim=c(0,1), psize=1, refline=coef(res), clim=c(0,1))
text(-1, 16, "Author(s)", pos=4)
```

```
text( 2, 16, "Proportion [95% CI]", pos=2)
```

---

fsn                            *Fail-Safe N Analysis (File Drawer Analysis)*

---

### Description

Function to compute the fail-safe N (also called file drawer analysis).

### Usage

```
fsn(yi, vi, sei, data, type="Rosenthal", alpha=.05, target, subset, digits=4)
```

### Arguments

| | |
|---|---|
| yi | vector with the observed effect sizes or outcomes. See 'Details' |
| vi | vector with the corresponding sampling variances. See 'Details' |
| sei | vector with the corresponding standard errors. See 'Details' |
| data | optional data frame containing the variables given to the arguments above. |
| type | character string indicating the method to use for the calculation of the fail-safe N. Possible options are "Rosenthal", "Orwin", or "Rosenberg". See below for more details. |
| alpha | target alpha level to use for the Rosenthal and Rosenberg methods (.05 by default). |
| target | target average effect size to use for the Orwin method. If undefined, then the target average effect size will be equal to the observed average effect size divided by 2. |
| subset | optional vector indicating the subset of studies that should be used for the calculation. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the observations to include. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (default is 4). |

### Details

The function can be used in conjunction with any of the usual effect size or outcome measures used in meta-analyses (e.g., log odds ratios, log relative risks, risk differences, mean differences, standardized mean differences, raw correlation coefficients, correlation coefficients transformed with Fisher's r-to-z transformation, and so on). Simply specify the observed outcomes via the yi argument and the corresponding sampling variances via the vi argument (instead of specifying vi, one can specify the standard errors (the square root of the sampling variances) via the sei argument). The [escalc](#) function can be used to compute a wide variety of effect size and outcome measures (and the corresponding sampling variances) based on summary statistics.

The Rosenthal method (sometimes called a 'file drawer analysis') calculates the number of studies averaging null results that would have to be added to the given set of observed outcomes to reduce

the combined significance level (p-value) to a target alpha level (e.g., .05). The calculation is based on Stouffer's method to combine p-values and is described in Rosenthal (1979).

The Orwin method calculates the number of studies averaging null results that would have to be added to the given set of observed outcomes to reduce the (unweighted) average effect size to a target (unweighted) average effect size. The method is described in Orwin (1983).

The Rosenberg method calculates the number of studies averaging null results that would have to be added to the given set of observed outcomes to reduce significance level (p-value) of the (weighted) average effect size (based on a fixed-effects model) to a target alpha level (e.g., .05). The method is described in Rosenberg (2005).

## Value

An object of class `"fsn"`. The object is a list containing the following components:

| | |
|---|---|
| type | the method used. |
| fsnum | the calculated fail-safe N. |
| alpha | the target alpha level. |
| pval | the p-value of the observed results. NA for the Orwin method. |
| meanes | the average effect size of the observed results. NA for the Rosenthal method. |
| target | the target effect size. NA for the Rosenthal and Rosenberg methods. |

The results are formated and printed with the [print.fsn](#) function.

## Note

For the Rosenberg method, the p-value is calculated based on a standard normal distribution (instead of a t-distribution, as suggested by Rosenberg, 2005).

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Rosenthal, R. (1979). The "file drawer problem" and tolerance for null results. *Psychological Bulletin*, **86**, 638–641.

Orwin, R. G. (1983). A fail-safe N for effect size in meta-analysis. *Journal of Educational Statistics*, **8**, 157–159.

Rosenberg, M. S. (2005). The file-drawer problem revisited: A general weighted method for calculating fail-safe numbers in meta-analysis. *Evolution*, **59**, 464–468.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

ranktest, regtest, trimfill

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

fsn(yi, vi, data=dat)
fsn(yi, vi, data=dat, type="Orwin")
fsn(yi, vi, data=dat, type="Rosenberg")
```

---

funnel                              *Funnel Plots for 'rma' Objects*

---

## Description

Function to create funnel plots for objects of class "rma".

## Usage

```
funnel(x, ...)

## S3 method for class 'rma'
funnel(x, yaxis="sei", xlim, ylim, xlab, ylab,
        steps=5, at, atransf, targs, digits,
        level=x$level, addtau2=FALSE, type="rstandard",
        back="lightgray", shade="white", hlines="white",
        refline, pch=19, pch.fill=21, ci.res=1000, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "rma". |
| yaxis | either "sei", "vi", "seinv", "vinv", "ni", "ninv", "sqrtni", "sqrtninv", "lni", or "wi" to indicate what values should be placed on the y-axis. See 'Details'. |
| xlim | x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| ylim | y-axis limits. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| ylab | title for the y-axis. If unspecified, the function tries to set an appropriate axis title. |
| steps | the number of tick marks for the y-axis (default is 5). |
| at | position of the x-axis tick marks and corresponding labels. If unspecified, the function tries to set the tick mark positions/labels to some sensible values. |

| atransf | optional argument specifying the name of a function that should be used to transform the x-axis labels (e.g., `transf=exp`). If unspecified, no transformation is used. |
|---|---|
| targs | optional arguments needed by the function specified via `atransf`. |
| digits | integer specifying the number of decimal places to which the tick mark labels of the x- and y-axis should be rounded. Can also be a vector of two integers, the first specifying the number of decimal places for the x-axis, the second for the y-axis labels (e.g., c(2,3)). If unspecified, the function tries to set the argument to some sensible values. |
| level | numerical value between 0 and 100 specifying the level of the pseudo confidence interval region (the default is to take the value from the object). May also be a vector of values to obtain multiple regions. See 'Examples'. |
| addtau2 | logical to indicate whether the amount of heterogeneity should be accounted for when drawing the pseudo confidence interval region (default is `FALSE`). Ignored when the model includes moderators and residuals are plotted. See 'Details'. |
| type | either `"rstandard"` (default) or `"rstudent"` indicating whether the usual or deleted residuals should be used in creating the funnel plot when the model involves moderators. See 'Details'. |
| back | color to use for the background of the plotting region. |
| shade | color to use for shading the pseudo confidence interval region. When `level` is a vector of values, different shading colors can be specified for each region. |
| hlines | color of the horizontal reference lines. |
| refline | value at which to draw the vertical reference line and, if drawn, where the pseudo confidence interval should be centered. If unspecified, the reference line is drawn at the fixed- or random-effects model estimate when the model does not include moderators and at zero when moderators are included (and therefore residuals are plotted). |
| pch | plotting symbol to use for the observed effect sizes or outcomes. By default, a solid circle is used. Can also be a vector of values. See [points](points) for other options. |
| pch.fill | plotting symbol to use for the effect sizes or outcomes filled in by the trim and fill method. By default, a circle is used. Only relevant when plotting an object created by the [trimfill](trimfill) function. |
| ci.res | integer specifying the number of y-axis values at which to calculate the bounds of the pseudo confidence interval. The default is `1000`, which usually provides a sufficient resolution for the plotting. |
| ... | other arguments. |

### Details

For fixed- and random-effects models (i.e., models not involving moderators), the plot shows the individual observed effect sizes or outcomes on the x-axis against the corresponding standard errors (i.e., the square root of the sampling variances) on the y-axis. A vertical line indicates the estimate based on the model. A pseudo confidence interval region is drawn around this value with bounds equal to $\pm 1.96 SE$, where $SE$ is the standard error value from the y-axis (assuming `level=95`). If `addtau2=TRUE`, then the bounds of the pseudo confidence interval region are equal to $\pm 1.96\sqrt{SE^2 + \tau^2}$, where $\tau^2$ is the amount of heterogeneity as estimated by the model.

For models involving moderators, the plot shows the residuals on the x-axis against their corresponding standard errors. Either the usual or deleted residuals can be used for that purpose (set via the type argument). See [residuals.rma](#) for more details on the different types of residuals.

Instead of placing the standard error value on the y-axis, several other options are available by setting the yaxis argument to:

- yaxis="vi" for the sampling variance,
- yaxis="seinv" for the inverse of the standard error,
- yaxis="vinv" for the inverse of the sampling variance,
- yaxis="ni" for the sample size,
- yaxis="ninv" for the inverse of the sample size,
- yaxis="sqrtni" for the square-root sample size,
- yaxis="sqrtninv" for the inverse of the square-root sample size,
- yaxis="lni" for the log of the sample size,
- yaxis="wi" for the weights.

However, only when yaxis="sei" (the default) will the pseudo confidence region have the expected (upside-down) funnel shape with straight lines. Also, when placing (a function of) the sample size on the y-axis or the weights, then the pseudo confidence region cannot be drawn. See Sterne and Egger (2001) for more details on the choice of the y-axis.

If the object passed to the function comes from the [trimfill](#) function, the effect sizes or outcomes that are filled in by the trim and fill method are also added to the funnel plot. The symbol to use for plotting the filled in values can then be specified via the pch.fill argument.

The arguments back, shade, and hlines can be set to NULL to suppress the shading and the horizontal reference lines.

## Value

A data frame with components:

| | |
|---|---|
| x | the x coordinates of the points that were plotted. |
| y | the y coordinates of the points that were plotted. |
| slab | study labels of the points that were plotted. |

Note that the data frame is returned invisibly.

## Note

Placing (a function of) the sample size on the y-axis (i.e., using yaxis="ni", yaxis="ninv", yaxis="sqrtni", yaxis="sqrtninv", or yaxis="lni") is only possible when information about the sample sizes is actually stored within the object passed to the funnel function. That should automatically be the case when the observed outcomes were computed with the escalc function or when the observed outcomes were computed within the model fitting function. On the other hand, this will not automatically be the case when rma.uni was used together with the yi and vi arguments and the yi and vi values were *not* computed with escalc. In that case, it is still possible to pass information about the sample sizes to the rma.uni function (i.e., use rma.uni(yi, vi, ni)).

When using unweighted estimation, using yaxis="wi" will place all points on a horizontal line.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Light, R. J., & Pillemer, D. B. (1984). *Summing up: The science of reviewing research*. Cambridge, MA: Harvard University Press.

Peters, J. L., Sutton, A. J., Jones, D. R., Abrams, K. R., & Rushton, L. (2008). Contour-enhanced meta-analysis funnel plots help distinguish publication bias from other causes of asymmetry. *Journal of Clinical Epidemiology*, **61**, 991–996.

Sterne, J. A. C., & Egger, M. (2001). Funnel plots for detecting bias in meta-analysis: Guidelines on choice of axis. *Journal of Clinical Epidemiology*, **54**, 1046–1055.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm, rma.mv, influence.rma.uni, trimfill

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### standard funnel plot
funnel(res)

### funnel plot with relative risk values on the x-axis
funnel(res, atransf=exp, at=log(c(.12, .25, .5, 1, 2)))

### contour-enhanced funnel plot centered at 0 (see Peters et al., 2008)
funnel(res, level=c(90, 95, 99), shade=c("white", "gray", "darkgray"),
       refline=0, atransf=exp, at=log(c(.10, .25, .5, 1, 2, 4, 10)))

### mixed-effects model with absolute latitude in the model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat,
           data=dat.bcg)

### funnel plot of the residuals
funnel(res)

### simulate a large meta-analytic dataset (correlations with rho = 0.2)
### with no heterogeneity or publication bias; then try out different
### versions of the funnel plot
```

```
gencor <- function(rhoi, ni) {
   x1 <- rnorm(ni, mean=0, sd=1)
   x2 <- rnorm(ni, mean=0, sd=1)
   x3 <- rhoi*x1 + sqrt(1-rhoi^2)*x2
   cor(x1, x3)
}

k  <- 200                              ### number of studies to simulate
ni <- round(rchisq(k, df=2) * 20 + 20)  ### simulate sample sizes (skewed distribution)
ri <- mapply(gencor, rep(0.2,k), ni)    ### simulate correlations

res <- rma(measure="ZCOR", ri=ri, ni=ni, method="FE") ### use r-to-z transformed correlations

funnel(res, yaxis="sei")
funnel(res, yaxis="vi")
funnel(res, yaxis="seinv")
funnel(res, yaxis="vinv")
funnel(res, yaxis="ni")
funnel(res, yaxis="ninv")
funnel(res, yaxis="sqrtni")
funnel(res, yaxis="sqrtninv")
funnel(res, yaxis="lni")
funnel(res, yaxis="wi")
```

---

hatvalues.rma.mv          *Hat values for 'rma.mv' Objects*

---

### Description

The function compute the hat values for objects of class "rma.mv".

### Usage

```
## S3 method for class 'rma.mv'
hatvalues(model, type="diagonal", ...)
```

### Arguments

| | |
|---|---|
| model | an object of class "rma.mv". |
| type | character string indicating whether to only return the diagonal of the hat matrix ("diagonal") or the entire matrix ("matrix"). |
| ... | other arguments. |

### Value

Either a vector with the diagonal elements of the hat matrix or the entire hat matrix.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., & Cheung, M. W.-L. (2010). Outlier and influence diagnostics for meta-analysis. *Research Synthesis Methods*, **1**, 112–125.

---

hc                              *Meta-Analysis based on the Method by Henmi and Copas (2010)*

---

## Description

The hc function can be used to obtain an estimate of the average true effect and corresponding confidence interval under a random-effects model using the method described by Henmi and Copas (2010).

## Usage

```
hc(object, ...)

## S3 method for class 'rma.uni'
hc(object, digits, transf, targs, control, ...)
```

## Arguments

| | |
|---|---|
| object | an object of class "rma.uni". |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| transf | optional argument specifying the name of a function that should be used to transform the estimated true effect and the corresponding interval bounds (e.g., transf=exp). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified under transf. |
| control | list of control values for the iterative algorithm. If unspecified, default values are defined inside the function. See 'Note'. |
| ... | other arguments. |

**Details**

The model specified by `object` must be a model without moderators (i.e., either a fixed- or a random-effects model and not a fixed-effects with moderators or mixed-effects model).

When using the usual method for fitting a random-effects model (i.e., weighted estimation with inverse-variance weights), the weights assigned to smaller and larger studies become more uniform as the amount of heterogeneity increases. As a consequence, the estimated average effect could become increasingly biased under certain forms of publication bias (where smaller studies on one side of the funnel plot are missing). The method by Henmi and Copas (2010) tries to counteract this problem by providing an estimate of the average true effect that is based on inverse-variance weights as used under a fixed-effects model (which do not take the amount of heterogeneity into consideration). The amount of heterogeneity is still estimated (with the DerSimonian-Laird estimator) and incorporated into the standard error of the estimated average effect and the corresponding confidence interval.

Currently, there is only a method for handling objects of class `"rma.uni"` with the `hc` function. It therefore provides a method for conducting a sensitivity analysis after the model has been fitted with the `rma.uni` function.

**Value**

An object of class `"hc.rma.uni"`. The object is a list containing the following components:

| | |
|---|---|
| b | estimated average true effect. |
| se | corresponding standard error. |
| ci.lb | lower bound of the confidence intervals for the average true effect. |
| ci.ub | upper bound of the confidence intervals for the average true effect. |
| ... | some additional elements/values. |

The results are formated and printed with the `print.hc.rma.uni` function.

**Note**

The method makes use of the `uniroot` function. By default, the desired accuracy is set equal to `.Machine$double.eps^0.25` and the maximum number of iterations to `1000`. The desired accuracy (`tol`) and the maximum number of iterations (`maxiter`) can be adjusted with the `control` argument (i.e., `control=list(tol=value, maxiter=value)`).

**Author(s)**

Original code by Henmi and Copas (2010). Corrected for typos by Michael Dewey (<info@aghmed.fsnet.co.uk>). Incorporated into the package with some small adjustments for consistency with the other functions in the package by Wolfgang Viechtbauer (<wvb@metafor-project.org>).

**References**

Henmi, M., & Copas, J. B. (2010). Confidence intervals for random effects meta-analysis and robustness to publication bias. *Statistics in Medicine*, **29**, 2969–2983.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

[rma.uni](#)

**Examples**

```
### load data from Lee & Done (2004)
data(dat.lee2004)

### meta-analysis based on log odds ratios
res <- rma(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat.lee2004)
res

### funnel plot as in Henmi and Copas (2010)
funnel(res, yaxis="seinv", refline=0, xlim=c(-3,3), ylim=c(.5,3.5), steps=7, digits=1)

### use method by Henmi and Copas (2010) as a sensitivity analysis
hc(res)

### back-transform results to odds ratio scale
hc(res, transf=exp)
```

---

influence.rma.uni            *Outlier and Influential Case Diagnostics for 'rma.uni' Objects*

---

**Description**

The functions compute various outlier and influential case diagnostics that indicate the influence of deleting one case/study at a time on the model fit and the fitted/residual values for objects of class "rma.uni".

**Usage**

```
## S3 method for class 'rma.uni'
influence(model, digits, ...)

## S3 method for class 'infl.rma.uni'
print(x, digits, ...)

## S3 method for class 'rma.uni'
cooks.distance(model, ...)
## S3 method for class 'rma.uni'
dfbetas(model, ...)

## S3 method for class 'rma.uni'
hatvalues(model, type="diagonal", ...)
```

**Arguments**

| | |
|---|---|
| `model` | an object of class `"rma.uni"`. |
| `x` | an object of class `"infl.rma.uni"` (for `print`). |
| `digits` | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| `type` | character string indicating whether to only return the diagonal of the hat matrix (`"diagonal"`) or the entire matrix (`"matrix"`). |
| `...` | other arguments. |

**Details**

The `influence` function calculates the following leave-one-out diagnostics for each study:

- externally standardized residual,
- DFFITS value,
- Cook's distance,
- covariance ratio,
- the leave-one-out amount of (residual) heterogeneity,
- the leave-one-out test statistic for the test of (residual) heterogeneity,
- DFBETAS value(s).

The diagonal elements of the hat matrix and the weights (in %) given to the observed effects or outcomes during the model fitting are also provided (except for their scaling, the hat values and weights are the same for models without moderators, but will differ when moderators are included).

For details on externally standardized residuals, see [rstudent.rma.uni](rstudent.rma.uni).

The DFFITS value essentially indicates how many standard deviations the predicted (average) effect for the $i$th study changes after excluding the $i$th study from the model fitting.

Cook's distance can be interpreted as the Mahalanobis distance between the entire set of predicted values once with the $i$th study included and once with the $i$th study excluded from the model fitting.

The covariance ratio is defined as the determinant of the variance-covariance matrix of the parameter estimates based on the dataset with the $i$th study removed divided by the determinant of the variance-covariance matrix of the parameter estimates based on the complete dataset. A value below 1 therefore indicates that removal of the $i$th study yields more precise estimates of the model coefficients.

The leave-one-out amount of (residual) heterogeneity is the estimated value of $\tau^2$ based on the dataset with the $i$th study removed. Note that this is always equal to 0 for fixed-effects models.

Similarly, the leave-one-out test statistic for the test of (residual) heterogeneity is the value of the test statistic of the test for (residual) heterogeneity calculated based on the dataset with the $i$th study removed.

Finally, the DFBETAS value(s) essentially indicate(s) how many standard deviations the estimated coefficient(s) change(s) after excluding the $i$th study from the model fitting.

A study may be considered to be 'influential' if at least one of the following is true:

- The absolute DFFITS value is larger than $3\sqrt{p/(k-p)}$, where $p$ is the number of model coefficients and $k$ the number of studies.
- The lower tail area of a chi-square distribution with $p$ degrees of freedom cut off by the Cook's distance is larger than 50%.
- The hat value is larger than $3(p/k)$.
- Any DFBETAS value is larger than $1$.

Studies which are considered influential with respect to any of these measures are marked with an asterisk. Note that the chosen cut-offs are (somewhat) arbitrary. Substantively informed judgment should always be used when examining the influence of each study on the results.

**Value**

An object of class `"infl.rma.uni"`. The object is a list containing the following components:

| | |
|---|---|
| `inf` | a data frame with columns equal to the externally standardized residuals, DFFITS values, Cook's distances, covariance ratios, leave-one-out $\tau^2$ estimates, leave-one-out (residual) heterogeneity test statistics, hat values, and weights. |
| `dfb` | a data frame with columns equal to the DFBETAS values. |
| `is.infl` | a logical vector indicating whether a particular study is considered to be 'influential'. |
| `...` | some additional elements/values. |

The results are printed with `print.infl.rma.uni` and plotted with `plot.infl.rma.uni`.

**Note**

Right now, the leave-one-out diagnostics are calculated by refitting the model $k$ times. Depending on how large $k$ is, it may take a few moments to finish the calculations. There are shortcuts for calculating at least some of these values without refitting the model each time, but these are currently not implemented (and may not exist for all of the leave-one-out diagnostics calculated by the function).

It may not be possible to fit the model after deletion of the $i$th study from the dataset. This will result in `NA` values for that study.

Certain relationships between the leave-one-out diagnostics and the (internally or externally) standardized residuals (Belsley, Kuh, & Welsch, 1980; Cook & Weisberg, 1982) no longer hold for the meta-analytic models. Maybe there are other relationships. These remain to be determined.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Belsley, D. A., Kuh, E., & Welsch, R. E. (1980). *Regression diagnostics*. New York: Wiley.

Cook, R. D., & Weisberg, S. (1982). *Residuals and influence in regression*. London: Chapman and Hall.

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. San Diego, CA: Academic Press.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., & Cheung, M. W.-L. (2010). Outlier and influence diagnostics for meta-analysis. *Research Synthesis Methods*, **1**, 112–125.

## See Also

plot.infl.rma.uni, rstudent.rma.uni, weights.rma.uni

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
influence(res)
plot(influence(res))

cooks.distance(res)
dfbetas(res)
hatvalues(res)
```

---

labbe                          *L'Abbe Plots for 'rma' Objects*

---

## Description

Function to create L'Abbé plots for objects of class "rma".

## Usage

```
labbe(x, ...)

## S3 method for class 'rma'
labbe(x, xlim, ylim, xlab, ylab,
      add=x$add, to=x$to, transf, targs,
      pch=21, psize, bg="gray", ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "rma". See 'Details'. |
| xlim | x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| ylim | y-axis limits. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| ylab | title for the y-axis. If unspecified, the function tries to set an appropriate axis title. |
| add | See below and the documentation of the [escalc](escalc) function for more details. |
| to | See below and the documentation of the [escalc](escalc) function for more details. |
| transf | optional argument specifying the name of a function that should be used to transform the outcomes (e.g., transf=exp). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified under transf. |
| pch | plotting symbol to use for the outcomes. By default, a filled circle is used. Can also be a vector of values. See [points](points) for other options. |
| psize | optional vector with point sizes for the outcomes. If unspecified, the point sizes are drawn proportional to the precision of the estimates. |
| bg | color to use for filling the plotting symbol (default is "gray"). Can also be a vector of values. Set to NA to make the plotting symbols transparent. |
| ... | other arguments. |

## Details

The model specified by x must be a model without moderators (i.e., either a fixed- or a random-effects model) fitted with either the [rma.uni](rma.uni), [rma.mh](rma.mh), [rma.peto](rma.peto), or [rma.glmm](rma.glmm) functions. Moreover, the model must be fitted with measure set equal to "RD" (for risk differences), "RR" (for relative risks), "OR" (for odds ratios), "AS" (for arcsine square-root transformed risk differences), "IRR" (for incidence rate ratios), "IRD" (for incidence rate differences), or "IRSD" (for square-root transformed incidence rate differences).

The function calculates the arm-level outcomes for the two experimental groups (e.g., treatment and control groups) and plots them against each other. In particular, the function plots the raw proportions of the two groups again each other when analyzing risk differences, the log of the proportions when analyzing (log) relative risks, the log odds when analyzing (log) odds ratios, the arcsine square-root transformed proportions when analyzing arcsine square-root transformed risk differences, the raw incidence rates when analyzing incidence rate differences, the log of the incidence rates when analyzing (log) incidence rate ratios, and the square-root transformed incidence rates when analyzing square-root transformed incidence rate differences. The transf argument can be used to transform these values (for example, transf=exp to transform the log of the proportions back to raw proportions).

As described under the documentation for the [escalc](escalc) function, zero cells can lead to problems when calculating particular outcomes. Adding a small constant to the cells of the $2 \times 2$ tables is a

common solution to this problem. By default, the functions adopts the same method for handling zero cells as was done when fitting the model.

The size of the points is drawn proportional to the precision (inverse standard error) of the outcomes. The solid line corresponds to identical outcomes in the two groups (i.e., the absence of a difference between the two groups). The dashed line indicates the estimated effect based on the fitted model.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

L'Abbé, K. A., Detsky, A. S., & O'Rourke, K. (1987). Meta-analysis in clinical research. *Annals of Internal Medicine*, **107**, 224–233.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### default plot
labbe(res)

### funnel plot with risk values on the x- and y-axis
labbe(res, transf=exp)
```

---

leave1out                          *Leave-One-Out Diagnostics for 'rma' Objects*

---

## Description

The functions repeatedly fit the specified model, leaving out one observation/study at a time.

## Usage

```
leave1out(x, ...)

## S3 method for class 'rma.uni'
leave1out(x, digits, transf, targs, ...)
## S3 method for class 'rma.mh'
leave1out(x, digits, transf, targs, ...)
## S3 method for class 'rma.peto'
leave1out(x, digits, transf, targs, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"rma.mh"`, `"rma.peto"`, or `"rma.uni"`. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| transf | an optional argument specifying the name of a function that should be used to transform the model coefficients and interval bounds (e.g., `transf=exp`). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified under `transf`. |
| ... | other arguments. |

## Details

The model specified by `x` must be a model without moderators (i.e., either a fixed- or a random-effects model).

## Value

An object of class `"list.rma"`. The object is a list containing the following components:

| | |
|---|---|
| estimate | estimated coefficients of the model. |
| se | standard errors of the coefficients. |
| zval | test statistics of the coefficients. |
| pval | p-values for the test statistics. |
| ci.lb | lower bounds of the confidence intervals for the coefficients. |
| ci.ub | upper bounds of the confidence intervals for the coefficients. |
| Q | test statistics for the tests of heterogeneity. |
| Qp | p-values for the tests of heterogeneity. |
| tau2 | estimated amounts of (residual) heterogeneity (only for random-effects models). |
| I2 | values of $I^2$ (only for random-effects models). |
| H2 | values of $H^2$ (only for random-effects models). |

The `"list.rma"` object is formated and printed with `print.list.rma`.

## Note

When using the `transf` option, the transformation is applied to the estimated coefficients and the corresponding interval bounds. The standard errors are then set equal to `NA` and are omitted from the printed output.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., & Cheung, M. W.-L. (2010). Outlier and influence diagnostics for meta-analysis. *Research Synthesis Methods*, **1**, 112–125.

## See Also

rma.uni, rma.mh, rma.peto

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res <- rma(yi, vi, data=dat)

### cumulative meta-analysis
leave1out(res)
leave1out(res, transf=exp)

### meta-analysis of the (log) relative risks using the Mantel-Haenszel method
res <- rma.mh(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### cumulative meta-analysis
leave1out(res)
leave1out(res, transf=exp)

### meta-analysis of the (log) odds ratios using Peto's method
res <- rma.peto(ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### cumulative meta-analysis
leave1out(res)
leave1out(res, transf=exp)
```

---

| | |
|---|---|
| llplot | *Log-Likelihood Plot of a Parameter Corresponding to an Outcome or Effect Size Measure* |

---

### Description

Function to plot the log-likelihood of a certain parameter corresponding to an outcome or effect size measure given the study data.

### Usage

```
llplot(measure="OR", ai, bi, ci, di, n1i, n2i, data, subset, drop00=TRUE,
       xvals=1000, xlim, ylim, xlab, ylab, scale=TRUE,
       lty, lwd, col, level=99.99, refline=0, ...)
```

### Arguments

| | |
|---|---|
| measure | a character string indicating for which effect size or outcome measure the log-likelihood should be calculated. See 'Details' for possible options and how the data should be specified. |
| ai | vector to specify the $2 \times 2$ table frequencies (upper left cell). |
| bi | vector to specify the $2 \times 2$ table frequencies (upper right cell). |
| ci | vector to specify the $2 \times 2$ table frequencies (lower left cell). |
| di | vector to specify the $2 \times 2$ table frequencies (lower right cell). |
| n1i | vector to specify the group sizes or row totals (first group/row). |
| n2i | vector to specify the group sizes or row totals (second group/row). |
| data | optional data frame containing the variables given to the arguments above. |
| subset | optional vector indicating the subset of studies that should be used. This can be a logical vector or a numeric vector indicating the indices of the studies to include. |
| drop00 | logical indicating whether studies with no cases (or only cases) in both groups should be dropped. See 'Details'. |
| xvals | integer specifying for how many distinct values of the (log) odds ratio the log-likelihood should be evaluated. |
| xlim | x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| ylim | y-axis limits. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| ylab | title for the y-axis. If unspecified, the function tries to set an appropriate axis title. |

| scale | logical indicating whether the log-likelihood values should be scaled, so that the total area under each curve is (approximately) equal to 1. |
| --- | --- |
| lty | the line type (either a single value or a vector of length $k$). If unspecified, the function uses solid lines for tables where the MLE of the odds ratio is finite and dashed/dotted lines otherwise. |
| lwd | the line width (either a single value or a vector of length $k$). If unspecified, the function sets the widths according to the sampling variances (so that the line is thicker for more precise studies and vice-versa). |
| col | the line color (either a single value or a vector of length $k$). If unspecified, the function uses various gray shades according to the sampling variances (so that darher shades are used for more precise studies and vice-versa). |
| level | numerical value between 0 and 100 specifying the plotting limits for each density line in terms of the confidence interval (default is 99.99). |
| refline | value at which a vertical 'reference' line should be drawn (default is 0). The line can be suppressed by setting this argument to NA. |
| ... | other arguments. |

## Details

At the moment, the function only accepts measure="OR". For each $2 \times 2$ table, the function then plots the log-likelihood of the (log) odds ratio based on the non-central hypergeometric distribution. Since studies with no cases (or only cases) in both groups have a flat likelihood and are not informative about the odds ratio, they are dropped by default (i.e., drop00=TRUE). For studies that have a single zero count, the MLE of the odds ratio is infinite and these likelihoods are indicated by a dashed line.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

van Houwelingen, H. C., Zwinderman, K. H., & Stijnen, T. (1993). A bivariate approach to meta-analysis. *Statistics in Medicine*, **12**, 2273–2284.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.glmm

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### create plot
llplot(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### load histamine H2 antagonist data
data(dat.collins1985a)

### create plot (Figure 2 in van Houwelingen, Zwinderman, & Stijnen, 1993)
llplot(measure="OR", ai=b.xci, n1i=nci, ci=b.xti, n2i=nti, data=dat.collins1985a,
       lwd=1, refline=NA, xlim=c(-4,4), drop00=FALSE)
```

---

metafor.news                   *Read News File of the Metafor Package*

---

## Description

Read news file of the **metafor-package**.

## Usage

```
metafor.news()
```

## Details

The function is just a wrapper for news(package="metafor") which parses and displays the 'NEWS' file of the package.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## Examples

```
metafor.news()
```

---

model.matrix.rma              *Model Matrix for 'rma' Objects*

---

### Description

The function extracts the model matrix for objects of class "rma".

### Usage

```
## S3 method for class 'rma'
model.matrix(object, ...)
```

### Arguments

object          an object of class "rma".

...             other arguments.

### Value

The model matrix.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

### References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

### See Also

[fitted.rma](fitted.rma)

### Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects meta-regression model
### with multiple moderators (absolute latitude, publication year, and allocation method)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
          mods = ~ ablat + year + alloc, data=dat.bcg)
model.matrix(res)
```

---

| permutest | *Permutation Tests for 'rma.uni' Objects* |

---

## Description

The function carries out permutation tests for objects of class `"rma.uni"`.

## Usage

```
permutest(x, ...)

## S3 method for class 'rma.uni'
permutest(x, exact=FALSE, iter=1000, progbar=TRUE,
          retpermdist=FALSE, digits, tol, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"rma.uni"`. |
| exact | logical indicating whether an exact permutation test should be carried out or not (default is `FALSE`). See 'Details'. |
| iter | integer specifying the number of iterations for the permutation test when not doing an exact test (default is 1000 iterations). |
| progbar | logical indicating whether a progress bar should be shown (default is `TRUE`). |
| retpermdist | logical indicating whether the permutation distributions of the test statistics should be returned (default is `FALSE`). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| tol | numerical tolerance used when making >= or <= comparisons between an observed test statistic and the test statistic under the permuted data. Defaults to `.Machine$double.eps^0.5` if unspecified. See 'Note'. |
| ... | other arguments. |

## Details

For models without moderators, the permutation test is carried out by permuting the signs of the observed effect sizes or outcomes. The (two-sided) p-value of the permutation test is then equal to two times the proportion of times that the test statistic under the permuted data is as extreme or more extreme than under the actually observed data. See Follmann and Proschan (1999) for more details.

For models with moderators, the permutation test is carried out by permuting the rows of the model matrix (i.e., $\mathbf{X}$). The (two-sided) p-value for a particular model coefficient is then equal to two times the proportion of times that the test statistic for the coefficient under the permuted data is as extreme or more extreme than under the actually observed data. Similarly, for the omnibus test, the p-value is the proportion of times that the test statistic for the omnibus test is as extreme or more

extreme than the actually observed one. See Higgins and Thompson (2004) and Viechtbauer et al. (in press) for more details.

If `exact=TRUE`, the function will try to carry out an exact permutation test. An exact permutation test requires fitting the model to each possible permutation once. However, the number of possible permutations increases rapidly with the number of outcomes/studies (i.e., $k$). For models without moderators, there are $2^k$ possible permutations of the signs. Therefore, for $k = 5$, there are 32 possible permutations, for $k = 10$, there are already 1024, and for $k = 20$, there are over one million permutations of the signs.

For models with moderators, the increase in the number of possible permutations may be even more severe. The total number of possible permutations of the model matrix is $k!$. Therefore, for $k = 5$, there are 120 possible permutations, for $k = 10$, there are 3,628,800, and for $k = 20$, there are over $10^{18}$ permutations of the model matrix.

Therefore, going through all possible permutations may become infeasible. Instead of using an exact permutation test, one can set `exact=FALSE` (which is also the default). In that case, the function approximates the exact permutation-based p-value(s) by going through a smaller number (as specified by the `iter` argument) of *random* permutations. Therefore, running the function twice on the same data will yield (slightly) different p-values. Setting `iter` sufficiently large ensures that the results become stable. Note that if `exact=FALSE` and `iter` is actually larger than the number of iterations required for an exact permutation test, then an exact test will be carried out.

For models with moderators, the exact permutation test actually only requires fitting the model to each *unique* permutation of the model matrix. The number of unique permutations will be smaller than $k!$ when the model matrix contains recurring rows. This may be the case when only including categorical moderators (i.e., factors) in the model or when any quantitative moderators included in the model only take on a small number of unique values. When `exact=TRUE`, the function therefore uses an algorithm to restrict the test to only the unique permutations of the model matrix, which may make the use of the exact test feasible even when $k$ is large.

When using random permutations, the function ensures that the very first permutation will always correspond to the original data. This avoids p-values equal to 0.

**Value**

An object of class `"permutest.rma.uni"`. The object is a list containing the following components:

| | |
|---|---|
| `pval` | p-value(s) based on the permutation test. |
| `QMp` | p-value for the omnibus test of coefficients based on the permutation test. |
| `zval.perm` | values of the test statistics of the coefficients under the various permutations (only when `retpermdist=TRUE`). |
| `QM.perm` | values of the test statistic for the omnibus test of coefficients under the various permutations (only when `retpermdist=TRUE`). |
| `...` | some additional elements/values are passed on. |

The results are formated and printed with the `print.permutest.rma.uni` function. One can also use `coef.permutest.rma.uni` to obtain the table with the model coefficients, corresponding standard errors, test statistics, p-values, and confidence interval bounds.

## Note

It is important to note that the p-values obtained with permutation tests cannot reach conventional levels of statistical significance (i.e., $p \leq .05$) when $k$ is very small. In particular, for models without moderators, the smallest possible (two-sided) p-value is .0625 when $k = 5$ and .03125 when $k = 6$. Therefore, the permutation test is only able to reject the null hypothesis at $\alpha = .05$ when $k$ is at least equal to 6. For models with moderators, the smallest possible (two-sided) p-value for a particular model coefficient is .0833 when $k = 4$ and .0167 when $k = 5$ (assuming that each row in the model matrix is unique). Therefore, the permutation test is only able to reject the null hypothesis at $\alpha = .05$ when $k$ is at least equal to 5.

When the number of permutations required for the exact test is so large as to be essentially indistinguishable from infinity (e.g., `factorial(200)`), the function will terminate with an error.

Determining whether a test statistic under the permuted data is as extreme or more extreme than under the actually observed data requires making >= or <= comparisons. To avoid problems due to the finite precision with which computers generally represent numbers, the function uses a numerical tolerance when making such comparisons (e.g., instead of `sqrt(3)^2 - 3 >= 0`, which may evaluate to `FALSE`, we can use `sqrt(3)^2 - 3 >= 0 - .Machine$double.eps^0.5`, which should evaluate to `TRUE`).

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Follmann, D. A., & Proschan, M. A. (1999). Valid inference in random effects meta-analysis. *Biometrics*, **55**, 732–737.

Good, P. I. (2009). *Permutation, parametric, and bootstrap tests of hypotheses* (3rd ed.). New York: Springer.

Higgins, J. P. T., & Thompson, S. G. (2004). Controlling the risk of spurious findings from meta-regression. *Statistics in Medicine*, **23**, 1663–1682.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., López-López, J. A., Sánchez-Meca, J., & Marín-Martínez, F. (in press). A comparison of procedures to test for moderators in mixed-effects meta-regression models. *Psychological Methods*.

## See Also

rma.uni, print.permutest.rma.uni

## Examples

```
### load BCG vaccine data
data(dat.bcg)
```

```
### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res <- rma(yi, vi, data=dat)

### permutation test (approximate and exact)
## Not run:

permutest(res)
permutest(res, exact=TRUE)

## End(Not run)

### mixed-effects model with two moderators (absolute latitude and publication year)
res <- rma(yi, vi, mods = ~ ablat + year, data=dat)

### permutation test (approximate only; exact not feasible)
## Not run:

permres <- permutest(res, iter=10000, retpermdist=TRUE)
permres

### histogram of permutation distribution for absolute latitude
### dashed horizontal line: the observed value of the test statistic
### red curve: standard normal density
### blue curve: kernel density estimate of the permutation distribution
### note that the tail area under the permutation distribution is larger
### than under a standard normal density (hence, the larger p-value)
hist(permres$zval.perm[,2], breaks=120, freq=FALSE, xlim=c(-5,5), ylim=c(0,.4),
     main="Permutation Distribution", xlab="Value of Test Statistic", col="gray90")
abline(v=res$zval[2], lwd=2, lty="dashed")
abline(v=0, lwd=2)
curve(dnorm, from=-5, to=5, add=TRUE, lwd=2, col=rgb(1,0,0,alpha=.7))
lines(density(permres$zval.perm[,2]), lwd=2, col=rgb(0,0,1,alpha=.7))

## End(Not run)
```

---

plot.cumul.rma                  *Plot Method for 'cumul.rma' Objects*

---

### Description

Plot method for objects of class "cumul.rma".

### Usage

```
## S3 method for class 'cumul.rma'
plot(x, yaxis="tau2", xlim, ylim, xlab, ylab,
     at, transf, atransf, targs, digits, cols=c("gray90","gray10"),
     addgrid=TRUE, pch=19, cex=1, lwd=2, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "cumul.rma". |
| yaxis | either "tau2", "I2", or "H2" to indicate what values should be placed on the y-axis. See 'Details'. |
| xlim | x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| ylim | y-axis limits. If unspecified, the function tries to set the y-axis limits to some sensible values. |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| ylab | title for the y-axis. If unspecified, the function tries to set an appropriate axis title. |
| at | position of the x-axis tick marks and corresponding labels. If unspecified, the function tries to set the tick mark positions/labels to some sensible values. |
| transf | optional argument specifying the name of a function that should be used to transform the summary estimates (e.g., transf=exp). If unspecified, no transformation is used. |
| atransf | optional argument specifying the name of a function that should be used to transform the x-axis labels (e.g., transf=exp). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified via transf or atransf. |
| digits | integer specifying the number of decimal places to which the tick mark labels of the x- and y-axis should be rounded. Can also be a vector of two integers, the first specifying the number of decimal places for the x-axis, the second for the y-axis labels (e.g., c(2,3)). If unspecified, the function tries to set the argument to some sensible values. |
| cols | vector with two or more colors to be used for indicating the order of the cumulative results. |
| addgrid | logical specifying whether a grid should be added to the plot (default is TRUE). |
| pch | plotting symbol to use. By default, a solid circle is used. See [points](#) for other options. |
| cex | symbol expansion factor. |
| lwd | line width. |
| ... | other arguments. |

## Details

The function is meant to be used for cumulative random-effects models results as obtained with the [cumul.rma.uni](#) function.

By default, the plot shows the model estimate on the x-axis and the corresponding estimate of $\tau^2$ on the y-axis in the cumulative order of the results in the "cumul.rma" object. Alternatively, one can also place the value of $I^2$ (yaxis="I2") or $H^2$ (yaxis="H2") on the y-axis.

The color gradient of the points/lines indicates the order of the cumulative results (by default, light gray at the beginning, dark gray at the end). A different set of colors to be used can be chosen via the cols argument. See 'Examples'.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[cumul.rma.uni](cumul.rma.uni)

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res <- rma(yi, vi, data=dat)

### cumulative meta-analysis (in the order of publication year)
sav <- cumul(res, transf=exp, order=order(dat$year))

### plot of model estimate and tau^2 over time
plot(sav)

### illustrate some other plot options
plot(sav, yaxis="I2", ylim=c(0,100), atransf=exp, at=log(seq(1.3, 1.6, by=.1)),
     lwd=5, cex=0, cols=c("green","blue","red"))
```

---

plot.infl.rma.uni          *Plot Method for 'infl.rma.uni' Objects*

---

## Description

Plot method for objects of class "infl.rma.uni".

## Usage

```
## S3 method for class 'infl.rma.uni'
plot(x, plotinf=TRUE, plotdfb=FALSE, dfbnew=FALSE, logcov=TRUE,
     layout, slab.style=1, las=0, pch=21, bg="black",
     bg.infl="red", col.na="lightgray", ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"infl.rma.uni"`. |
| plotinf | logical indicating whether the various case diagnostics should be plotted (default is TRUE). Can also be a vector of up to 8 integers indicating which plots to draw (see 'Details' for the numbers corresponding to the various plots). |
| plotdfb | logical indicating whether the DFBETAS values should be plotted (default is FALSE). Can also be a vector of integers to indicate for which coefficient(s) to plot the DFBETAS values. |
| dfbnew | logical indicating whether a new device should be opened for plotting the DF-BETAS values (default is FALSE). |
| logcov | logical indicating whether the covariance ratios should be plotted on a log scale (default is TRUE). |
| layout | optional vector of two numbers, indicating the number of rows and columns for the layout of the figure. |
| slab.style | integer to indicate the style of the x-axis labels: 1 = study number, 2 = study label, 3 = abbreviated study label. Note that study labels, even when abbreviated, may be too long to fit in the margin.) |
| las | integer between 0 and 3 to specify the alignment of the axis labels (see [par](par)). The most useful alternative to 0 is 3, so that the x-axis labels are drawn vertical to the axis. |
| pch | plotting symbol to use. By default, a filled circle is used. See [points](points) for other options. |
| bg | color to use for filling the plotting symbol (default is `"black"`). |
| bg.infl | color to use for filling the plotting symbol when the point is considered influential (default is `"red"`). |
| col.na | color to use for lines connecting two points with NA values in between (default is `"lightgray"`). |
| ... | other arguments. |

## Details

When `plotinf=TRUE`, the function plots the (1) externally standardized residuals, (2) DFFITS values, (3) Cook's distances, (4) covariance ratios, (5) leave-one-out $\tau^2$ estimates, (6) leave-one-out (residual) heterogeneity test statistics, (7) hat values, and (8) weights. If `plotdfb=TRUE`, the DF-BETAS values are also plotted either after confirming the page change (if `newdfb=FALSE`) or on a separate device (if `newdfb=TRUE`).

A study may be considered to be 'influential' if at least one of the following is true:

- The absolute DFFITS value is larger than $3\sqrt{p/(k-p)}$, where $p$ is the number of model coefficients and $k$ the number of studies.
- The lower tail area of a chi-square distribution with $p$ degrees of freedom cut off by the Cook's distance is larger than 50%.
- The hat value is larger than $3(p/k)$.
- Any DFBETAS value is larger than $1$.

Studies which are considered influential with respect to any of these measures are indicated by the color specified for the `bg.infl` argument (the default is `"red"`).

The cut-offs described above are indicated in the plot with horizontal reference lines. In addition, on the plot of the externally standardized residuals, horizontal reference lines are drawn at -1.96, 0, and 1.96. On the plot of the hat values, a horizontal reference line is drawn at $p/k$. Since the sum of the hat values is equal to $p$, the value $p/k$ indicates equal hat values for all $k$ studies. Finally, on the plot of weights, a horizontal reference line is drawn at $100/k$, corresponding to the value for equal weights (in %) for all $k$ studies. Note that all weights will automatically be equal to each other when using unweighted model fitting. Also, the hat values will be equal to the weights values (except for their scaling) in models without moderators.

The chosen cut-offs are (somewhat) arbitrary. Substantively informed judgment should always be used when examining the influence of each study on the results.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., & Cheung, M. W.-L. (2010). Outlier and influence diagnostics for meta-analysis. *Research Synthesis Methods*, **1**, 112–125.

### See Also

influence.rma.uni

### Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
plot(influence(res))
plot(influence(res), plotinf=1:4)
plot(influence(res), plotinf=1:4, layout=c(4,1))
plot(influence(res), plotdfb=TRUE)
```

---

plot.rma.uni                    *Plot Method for 'rma' Objects*

---

### Description

Plot method for objects of class ″rma.uni″, ″rma.mh″, ″rma.peto″, and ″rma.glmm″.

### Usage

```
## S3 method for class 'rma.uni'
plot(x, qqplot=FALSE, ...)

## S3 method for class 'rma.mh'
plot(x, qqplot=FALSE, ...)

## S3 method for class 'rma.peto'
plot(x, qqplot=FALSE, ...)

## S3 method for class 'rma.glmm'
plot(x, qqplot=FALSE, ...)
```

### Arguments

| | |
|---|---|
| x | an object of class ″rma.uni″, ″rma.mh″, or ″rma.peto″. The method is not yet implemented for objects of class ″rma.glmm″. |
| qqplot | logical indicating whether a normal QQ plot should be drawn (default is FALSE). |
| ... | other arguments. |

### Details

Four plots are produced. If the model does not contain any moderators, then a forest plot, funnel plot, radial plot, and a plot of the standardized residuals is provided. If qqplot=TRUE, the last plot is replaced by a normal QQ plot of the standardized residuals.

If the model contains moderators, then a forest plot, funnel plot, plot of the standardized residuals against the fitted values, and a plot of the standardized residuals is provided. If qqplot=TRUE, the last plot is replaced by a normal QQ plot of the standardized residuals.

### Note

If the number of studies is large, the forest plot may become quite impossible to read due to the small font size. Stretching the plotting device vertically should provide more room.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. `http://www.jstatsoft.org/v36/i03/`.

## See Also

`forest`, `funnel`, `radial`, `qqnorm.rma.uni`

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
plot(res, qqplot=TRUE)

### mixed-effects model with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
plot(res, qqplot=TRUE)
```

---

predict.rma                  *Predicted Values for 'rma' Objects*

---

## Description

The function computes predicted values, corresponding standard errors, confidence intervals, and (approximate) credibility/prediction intervals for objects of class `"rma"`.

## Usage

```
## S3 method for class 'rma'
predict(object, newmods, intercept, tau2.levels, addx=FALSE,
        level, digits, transf, targs, ...)
```

## Arguments

| | |
|---|---|
| `object` | an object of class `"rma"`. |
| `newmods` | optional vector or matrix specifying the values of the moderator values for which the predicted values should be calculated. See 'Details'. |
| `intercept` | logical specifying whether the intercept term should be included when calculating the predicted values for `newmods`. If unspecified, the intercept is automatically added when the original model also included an intercept. |
| `tau2.levels` | vector specifying the levels of the inner factor when computing credibility/prediction intervals. Only relevant for models of class `"rma.mv"` (see `rma.mv`) and when the model includes more than a single $\tau^2$ value. See 'Details'. |

addx          logical specifying whether the values of the moderator variables should be added
              to the returned object. See 'Examples'.

level         numerical value between 0 and 100 specifying the confidence and credibil-
              ity/prediction interval level (if unspecified, the default is to take the value from
              the object).

digits        integer specifying the number of decimal places to which the printed results
              should be rounded (if unspecified, the default is to take the value from the ob-
              ject).

transf        optional argument specifying the name of a function that should be used to trans-
              form the predicted values and interval bounds (e.g., transf=exp). If unspeci-
              fied, no transformation is used.

targs         optional arguments needed by the function specified under transf.

...           other arguments.

### Details

For a fixed-effects model, predict(object) returns the estimated (average) outcome in the set of
studies included in the meta-analysis. This is the same as the estimated intercept in the fixed-effects
model.

For a random-effects model, predict(object) returns the estimated (average) outcome in the
hypothetical population of studies from which the set of studies included in the meta-analysis are
assumed to be a random selection. This is the same as the estimated intercept in the random-effects
model.

For models including one or more moderators, predict(object) returns the estimated (average)
outcomes for values of the moderator(s) equal to those of the $k$ studies included in the meta-analysis
(i.e., the fitted values for the $k$ studies).

For models including $q$ moderator variables, new moderator values for $k_{new}$ new studies can be
specified by setting newmods equal to an $k_{new} \times q$ matrix with the corresponding new moderator
values. If the original model includes an intercept, then it should not be explicitly specified under
newmods, as it will be added by default (unless one sets intercept=FALSE). Also, any factors in
the original model get turned into the appropriate contrast variables within the rma function, so that
newmods should actually include the values for the contrast variables. Examples are shown below.

For random/mixed-effects models, an approximate credibility/prediction interval is also calculated
(Raudenbush, 2009; Riley et al., 2011). The interval estimates where level % of the true out-
comes fall in the hypothetical population of studies. Note that this interval is calculated under the
assumption that the value of $\tau^2$ is known (and not estimated). A method for calculating a credibil-
ity/prediction interval that accounts for the uncertainty in the estimate of $\tau^2$ may be implemented
in the future.

For random-effects models of class "rma.mv" (see [rma.mv](rma.mv)) with multiple $\tau^2$ values (i.e., when
the random argument includes a ~ inner | outer term and struct="HCS", struct="HAR", or
struct="UN"), the function will provide credibility/prediction intervals for each level of the inner
factor (since the credibility/prediction intervals differ depending on the $\tau^2$ value). Optionally, one
can use the tau2.levels argument to specify for which level(s) the credibility/prediction interval
should be provided.

When using the newmods argument for mixed-effects models of class "rma.mv" with multiple $\tau^2$ values, one must use the tau2.levels argument to specify the levels of the inner factor (i.e., a vector of length $k_{new}$) to obtain the appropriate credibility/prediction interval(s).

**Value**

An object of class "list.rma". The object is a list containing the following components:

| | |
|---|---|
| pred | predicted value(s). |
| se | corresponding standard error(s). |
| ci.lb | lower bound of the confidence interval(s). |
| ci.ub | upper bound of the confidence interval(s). |
| cr.lb | lower bound of the credibility/prediction interval(s) (only random/mixed-effects models). |
| cr.ub | upper bound of the credibility/prediction interval(s) (only random/mixed-effects models). |
| tau2.level | the level(s) of the inner factor (only for models of class "rma.mv" with multiple $\tau^2$ values). |
| X | the moderator value(s) used to calculate the predicted values (only when addx=TRUE). |
| ... | some additional elements/values. |

The "list.rma" object is formated and printed with print.list.rma.

**Note**

The predicted values are based only on the fixed effects of the model. Best linear unbiased predictions (BLUPs) that combine the fitted values based on the fixed effects and the estimated contributions of the random effects can be obtained with blup.rma.uni (currently only for objects of class "rma.uni").

When using the transf option, the transformation is applied to the predicted values and the corresponding interval bounds. The standard errors are then set equal to NA and are omitted from the printed output.

Confidence and credibility/prediction intervals are calculated based on the critical values from a standard normal distribution (i.e., $\pm 1.96$ for level=95). When the model was fitted with the Knapp and Hartung (2003) method (i.e., knha=TRUE in the rma.uni function or tdist=TRUE in the rma.glmm and rma.mv functions), then the t-distribution with $k - p$ degrees of freedom is used.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. San Diego, CA: Academic Press.

Raudenbush, S. W. (2009). Analyzing effect sizes: Random effects models. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 295–315). New York: Russell Sage Foundation.

Riley, R. D., Higgins, J. P., & Deeks, J. J. (2011). Interpretation of random effects meta-analyses. *British Medical Journal*, **342**, d549.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

fitted.rma, blup.rma.uni

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### average relative risk with 95% CI
predict(res, transf=exp)

### mixed-effects model with absolute latitude as a moderator
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat,
           data=dat.bcg)

### predicted average relative risks for given absolute latitude values
predict(res, transf=exp, addx=TRUE)

### predicted average relative risks for 10-60 degrees absolute latitude
predict(res, newmods=c(10, 20, 30, 40, 50, 60), transf=exp)

### mixed-effects model with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)

### predicted average relative risks for 10 and 60 degrees latitude in 1950 and 1980
predict(res, newmods=cbind(c(10,60,10,60),c(1950,1950,1980,1980)), transf=exp, addx=TRUE)

### mixed-effects model with two moderators (one of which is a factor)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + factor(alloc), data=dat.bcg)

### examine how the factor was actually coded for the studies in the dataset
predict(res, addx=TRUE)
```

```
### predictd average relative risks at 30 degrees for the three factor levels
### note: the contrast (dummy) variables need to specified explicitly here
predict(res, newmods=c(30, 0, 0), addx=TRUE)   # for alternate  allocation
predict(res, newmods=c(30, 1, 0), addx=TRUE)   # for random     allocation
predict(res, newmods=c(30, 0, 1), addx=TRUE)   # for systematic allocation
```

---

print.anova.rma *Print Method for 'anova.rma' Objects*

---

### Description

Print method for objects of class "anova.rma".

### Usage

```
## S3 method for class 'anova.rma'
print(x, digits, ...)
```

### Arguments

x           an object of class "anova.rma".

digits      integer specifying the number of decimal places to which the printed results
            should be rounded (if unspecified, the default is to take the value from the ob-
            ject).

...         other arguments.

### Details

The output includes:

- the number of parameters in the full and the reduced model.
- the AIC, BIC, AICc, and log-likelihood of the full and the reduced model.
- the value of the likelihood ratio test statistic.
- the p-value for the likelihood ratio test.
- the test statistic for the test of (residual) heterogeneity for the full and the reduced model.
- the estimate of $\tau^2$ from the full and the reduced model. Suppressed for fixed-effects models.
- the amount of (residual) heterogeneity in the reduced model that is accounted for in the full model (in percent). NA for fixed-effects models or if the amount of heterogeneity in the reduced model is equal to zero. This can be regarded as a pseudo $R^2$ statistic (Raudenbush, 2009).

The last two items are not provided when comparing "rma.mv" models.

### Value

The function does not return an object.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Raudenbush, S. W. (2009). Analyzing effect sizes: Random effects models. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 295–315). New York: Russell Sage Foundation.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[anova.rma](anova.rma)

---

print.confint.rma  *Print Method for 'confint.rma' Objects*

---

## Description

Print method for objects of class "confint.rma".

## Usage

```
## S3 method for class 'confint.rma'
print(x, digits, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "confint.rma". |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

## Details

The output includes:

- estimate of the model coefficient or variance component
- lower bound of the confidence interval
- upper bound of the confidence interval

## Value

The function does not return an object.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[confint.rma.uni](confint.rma.uni)

---

print.escalc          *Print and Summary Methods for 'escalc' Objects*

---

## Description

Print and summary methods for objects of class `"escalc"`.

## Usage

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

## S3 method for class 'escalc'
summary(object, out.names=c("sei","zi","ci.lb","ci.ub"), var.names,
        H0=0, append=TRUE, replace=TRUE, level=95, digits, transf, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"escalc"`. |
| object | an object of class `"escalc"`. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (the default is to take the value from the object if possible). |
| out.names | character string with four elements, specifying the variable names for the standard errors, test statistics, and lower/upper confidence interval bounds. |
| var.names | character string with two elements, specifying the variable names for the observed outcomes and the sampling variances (the default is to take the value from the object if possible). |
| H0 | numeric value specifying the value of the outcome under the null hypothesis. |

| append | logical indicating whether the data frame specified via the `object` argument should be returned together with the additional variables that are calculated by the `summary` function (default is `TRUE`). |
|---|---|
| replace | logical indicating whether existing values for `sei`, `zi`, `ci.lb`, and `ci.ub` in the data frame should be replaced or not. Only relevant when the data frame already contains these variables. If `replace=TRUE` (the default), all of the existing values will be overwritten. If `replace=FALSE`, only NA values will be replaced. |
| level | numerical value between 0 and 100 specifying the confidence interval level (the default is 95). |
| transf | optional argument specifying the name of a function that should be used to transform the observed outcomes and interval bounds (e.g., `transf=exp`). If unspecified, no transformation is used. Any additional arguments needed for the function specified here can be passed via `...`. |
| ... | other arguments. |

## Value

The `print.escalc` function formats and prints the data frame, so that the observed outcomes and sampling variances are rounded (to the number of digits specified).

The `summary.escalc` function creates an object that is a data frame containing the original data (if `append=TRUE`) and the following components:

| yi | observed outcomes or effect size estimates (transformed if `transf` is not `FALSE`). |
|---|---|
| vi | corresponding (estimated) sampling variances. |
| sei | standard errors of the observed outcomes or effect size estimates. |
| zi | test statistics for testing $H_0 : \theta_i = H0$ (i.e., `(yi-H0)/sei`). |
| ci.lb | lower confidence interval bounds (transformed if `transf` is not `FALSE`). |
| ci.ub | upper confidence interval bounds (transformed if `transf` is not `FALSE`). |

Note that the actual variable names above depend on the `out.names` (and `var.names`) arguments. If the data frame already contains variables with names as specified by the `out.names` argument, the values for these variables will be overwritten when `replace=TRUE` (which is the default). By setting `replace=FALSE`, only values that are NA will be replaced.

The `print.escalc` function again formats and prints the data frame, rounding the added variables to the number of digits specified.

## Note

If some transformation function has been specified for the `transf` argument, then `yi`, `ci.lb`, and `ci.ub` will be transformed accordingly. However, `vi` and `sei` then still reflect the sampling variances and standard errors of the untransformed values.

The `summary.escalc` function computes `level` % Wald-type confidence intervals, which may or may not be the most accurate method for computing confidence intervals for the chosen outcome or effect size measure.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

escalc

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
dat

### apply summary function
summary(dat)
summary(dat, transf=exp)
```

---

print.fsn                   *Print Method for 'fsn' Objects*

---

## Description

Print method for objects of class "fsn".

## Usage

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

## Arguments

| | |
|---|---|
| x | an object of class "fsn". |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

## Details

The output shows the results from the fail-safe N calculation.

## Value

The function does not return an object.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[fsn](#)

---

| print.hc.rma.uni | *Print Method for 'hc.rma.uni' Objects* |
|---|---|

---

## Description

Print method for objects of class ″hc.rma.uni″.

## Usage

```
## S3 method for class 'hc.rma.uni'
print(x, digits, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class ″hc.rma.uni″. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

**Details**

The output is a data frame with two rows, the first (labeled rma) corresponding to the results based on the usual estimation method, the second (labeled hc) corresponding to the results based on the method by Henmi and Copas (2010). The data frame includes the following variables:

- the method used to estimate $\tau^2$ (always DL for hc)

- the estimated amount of heterogeneity

- the estimated average true effect

- the corresponding standard errror (NA when transf argument has been used)

- the lower and upper confidence interval bounds

**Value**

The function returns the data frame invisibly.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

**References**

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

hc.rma.uni

---

print.list.rma                    *Print Method for 'list.rma' Objects*

---

**Description**

Print method for objects of class "list.rma".

**Usage**

```
## S3 method for class 'list.rma'
print(x, digits, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"list.rma"`. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

## Value

See the documentation of the function that creates the `"list.rma"` object for details on what is printed.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

---

print.permutest.rma.uni

*Print Method for 'permutest.rma.uni' Objects*

---

## Description

Print method for objects of class `"permutest.rma.uni"`.

## Usage

```
## S3 method for class 'permutest.rma.uni'
print(x, digits, signif.stars=getOption("show.signif.stars"),
      signif.legend=signif.stars, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"permutest.rma.uni"`. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| signif.stars | logical indicating whether p-values should be encoded visually with 'significance stars'. Defaults to the `show.signif.stars` slot of `options`. |
| signif.legend | logical indicating whether the legend for the 'significance stars' should be printed. Defaults to the value for `signif.stars`. |
| ... | other arguments. |

**Details**

The output includes:

- the results of the omnibus test of the coefficients in the model. Suppressed if the model includes only one coefficient (e.g., only an intercept, like in the fixed- and random-effects model). The p-value based on the permutation test is indicated by `p-val*`.
- a table with the estimated coefficients, corresponding standard errors, test statistics, p-values, and confidence interval bounds. The p-values based on the permutation test are indicated by `pval*`.

**Value**

The function does not return an object.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

**References**

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

**See Also**

[permutest.rma.uni](permutest.rma.uni)

---

print.ranktest.rma          *Print Method for 'ranktest.rma' Objects*

---

**Description**

Print method for objects of class `"ranktest.rma"`.

**Usage**

```
## S3 method for class 'ranktest.rma'
print(x, digits, ...)
```

**Arguments**

| | |
|---|---|
| x | an object of class `"ranktest.rma"`. |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

## Details

The output includes:

- the estimated value of Kendall's tau rank correlation coefficient
- the corresponding p-value for the test that the true tau is equal to zero

## Value

The function does not return an object.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

ranktest.rma

---

print.regtest.rma          *Print Method for 'regtest.rma' Objects*

---

## Description

Print method for objects of class "regtest.rma".

## Usage

```
## S3 method for class 'regtest.rma'
print(x, digits, ret.fit, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "regtest.rma". |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ret.fit | logical indicating whether the full results from the fitted model should also be returned (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

**Details**

The output includes:

- the model used for the regression test
- the predictor used for the regression test
- the results from the fitted model (only when `ret.fit=TRUE`)
- the value of the test statistic for the test that the predictor is unreleated to the outcomes
- the degrees of freedom of the test statistic (only if the test statistic follows a t-distribution)
- the p-value for the test statistic

**Value**

The function does not return an object.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

**References**

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

**See Also**

[regtest.rma](regtest.rma)

---

print.rma.uni                *Print and Summary Methods for 'rma' Objects*

---

**Description**

Print and summary methods for objects of class `"rma.uni"`, `"rma.mh"`, `"rma.peto"`, `"rma.glmm"`, and `"rma.glmm"`.

**Usage**

```
## S3 method for class 'rma.uni'
print(x, digits, showfit=FALSE, signif.stars=getOption("show.signif.stars"),
      signif.legend=signif.stars, ...)

## S3 method for class 'rma.mh'
print(x, digits, showfit=FALSE, ...)
```

```
## S3 method for class 'rma.peto'
print(x, digits, showfit=FALSE, ...)

## S3 method for class 'rma.glmm'
print(x, digits, showfit=FALSE, signif.stars=getOption("show.signif.stars"),
      signif.legend=signif.stars, ...)

## S3 method for class 'rma.mv'
print(x, digits, showfit=FALSE, signif.stars=getOption("show.signif.stars"),
      signif.legend=signif.stars, ...)

## S3 method for class 'rma'
summary(object, digits, showfit=TRUE, ...)

## S3 method for class 'summary.rma'
print(x, digits, showfit=TRUE, signif.stars=getOption("show.signif.stars"),
      signif.legend=signif.stars, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class "rma.uni", "rma.mh", "rma.peto", "rma.glmm", "rma.mv", or "summary.rma" (for print). |
| object | an object of class "rma" (for summary). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| showfit | logical indicating whether the fit statistics and information criteria should be printed (default is FALSE for print and TRUE for summary). |
| signif.stars | logical indicating whether p-values should be encoded visually with 'significance stars'. Defaults to the show.signif.stars slot of [options](). |
| signif.legend | logical indicating whether the legend for the 'significance stars' should be printed. Defaults to the value for signif.stars. |
| ... | other arguments. |

## Details

The output includes:

- the log-likelihood, deviance, AIC, BIC, and AICc value (when setting showfit=TRUE or by default for summary).

- for objects of class "rma.uni" and "rma.glmm", the amount of (residual) heterogeneity in the random/mixed-effects model (i.e., the estimate of $\tau^2$ and its square root). Suppressed for fixed-effects models. The (asymptotic) standard error of the estimate of $\tau^2$ is also provided (where possible).

- for objects of "rma.mv", a table providing information about the variance components and correlations in the model. For $\sigma^2$ components, the estimate and its square root are provided, in addition to the number of values/levels, whether the component was fixed or estimated, and

the name of the grouping variable/factor. If the R argument was used to specify known corre-
lation matrices, this is also indicated. For models with an `~ inner | outer` formula term, the
name of the inner and outer grouping variable/factor are given and the number of values/levels
of these variables/factors. In addition, for each $\tau^2$ component, the estimate and its square root
are provided, the number of effects or outcomes observed at each level of the inner group-
ing variable/factor (only for `struct="HCS"`, `struct="HAR"`, and `struct="UN"`), and whether
the component was fixed or estimated. Finally, either the estimate of $\rho$ (for `struct="CS"`,
`struct="AR"`, `struct="HAR"`, or `struct="HCS"`) or the entire estimated correlation matrix
(for `struct="UN"`) between the levels of the inner grouping variable/factor is provided, again
with information whether a particular correlation was fixed or estimated, and how often each
combination of levels of the inner grouping variable/factor was observed across the levels of
the outer grouping variable/factor.

- for objects of class `"rma.uni"` and `"rma.glmm"`, the $I^2$ statistic. For a random-effects model,
  $I^2$ estimates (in percent) how much of the total variability in the effect size estimates (which
  is composed of heterogeneity plus sampling variability) can be attributed to heterogeneity
  among the true effects. For a mixed-effects model, $I^2$ estimates how much of the unaccounted
  variability (which is composed of residual heterogeneity plus sampling variability) can be
  attributed to residual heterogeneity. See 'Note' for how $I^2$ is computed.

- for objects of class `"rma.uni"` and `"rma.glmm"`, the $H^2$ statistic. For a random-effects model,
  $H^2$ estimates the ratio of the total amount of variability in the effect size estimates to the
  amount of sampling variability. For a mixed-effects model, $H^2$ estimates the ratio of the
  unaccounted variability in the effect size estimates to the amount of sampling variability. See
  'Note' for how $H^2$ is computed.

- for objects of class `"rma.uni"`, the $R^2$ statistic, which estimates the amount of heterogeneity
  accounted for by the moderators included in the model and can be regarded as a pseudo $R^2$
  statistic (Raudenbush, 2009). Only provided when fitting a mixed-effects models (i.e., for
  models including moderators). This is suppressed (and set to NULL) for models without mod-
  erators, fixed-effects models, or if the model does not contain an intercept. NA if the amount
  of heterogeneity is equal to zero to begin with. See 'Note' for how $R^2$ is computed.

- for objects of class `"rma.glmm"`, the amount of study level variability (only when using a
  model that models study level differences as a random effect).

- the results of the test for (residual) heterogeneity. This is the usual $Q$-test for heterogeneity
  when not including moderators in the model and the $Q_E$-test for residual heterogeneity when
  moderators are included. For objects of class `"rma.glmm"`, the results from a Wald-type test
  and a likelihood ratio test are provided (see `rma.glmm` for more details).

- the results of the omnibus (Wald-type) test of the coefficients in the model (the indices of the
  coefficients tested are also indicated). Suppressed if the model includes only one coefficient
  (e.g., only an intercept, like in the fixed- and random-effects model).

- a table with the estimated coefficients, corresponding standard errors, test statistics, p-values,
  and confidence interval bounds.

- the Cochran-Mantel-Haenszel test and Tarone's test for heterogeneity (only when analyzing
  odds ratios using the Mantel-Haenszel method, i.e., `"rma.mh"`).

**Value**

The `print` functions do not return an object. The `summary` function returns the object passed to it
(with additional class `"summary.rma"`).

## Note

For random-effects models, the $I^2$ statistic is computed with

$$I^2 = 100\% \times \frac{\hat{\tau}^2}{\hat{\tau}^2 + s^2},$$

where $\hat{\tau}^2$ is the estimated value of $\tau^2$ and

$$s^2 = \frac{\sum w_i(k-1)}{(\sum w_i)^2 - \sum w_i^2},$$

where $w_i$ is the inverse of the sampling variance of the $i$th study ($s^2$ is equation 9 in Higgins & Thompson, 2002, and can be regarded as the 'typical' within-study variance of the observed effect sizes or outcomes). The $H^2$ statistic is computed with

$$H^2 = \frac{\hat{\tau}^2 + s^2}{s^2}.$$

Analogous equations are used for mixed-effects models.

Therefore, depending on the estimator of $\tau^2$ used, the values of $I^2$ and $H^2$ will change. For random-effects models, $I^2$ and $H^2$ are typically computed in practice with $I^2 = (Q - (k - 1))/Q$ and $H^2 = Q/(k - 1)$, where $Q$ denotes the statistic for the test of heterogeneity and $k$ the number of studies (i.e., observed effects or outcomes) included in the meta-analysis. The equations used in the metafor package to compute these statistics are more general and have the advantage that the values of $I^2$ and $H^2$ will be consistent with the estimated value of $\tau^2$ (i.e., if $\hat{\tau}^2 = 0$, then $I^2 = 0$ and $H^2 = 1$ and if $\hat{\tau}^2 > 0$, then $I^2 > 0$ and $H^2 > 1$).

The two definitions of $I^2$ and $H^2$ actually coincide when using the DerSimonian-Laird estimator of $\tau^2$ (i.e., the commonly used equations are actually special cases of the more general definitions given above). Therefore, if you prefer the more conventional definitions of these statistics, use method="DL" when fitting the random/mixed-effects model with the `rma.uni` function.

The pseudo $R^2$ statistic (Raudenbush, 2009) is computed with

$$R^2 = \frac{\hat{\tau}^2_{RE} - \hat{\tau}^2_{ME}}{\hat{\tau}^2_{RE}},$$

where $\hat{\tau}^2_{RE}$ denotes the estimated value of $\tau^2$ based on the random-effects model (i.e., the total amount of heterogeneity) and $\hat{\tau}^2_{ME}$ denotes the estimated value of $\tau^2$ based on the mixed-effects model (i.e., the residual amount of heterogeneity). It can happen that $\hat{\tau}^2_{RE} < \hat{\tau}^2_{ME}$, in which case $R^2$ is set to zero. Again, the value of $R^2$ will change depending on the estimator of $\tau^2$ used. Also note that this statistic is only computed when the mixed-effects model includes an intercept (so that the random-effects model is clearly nested within the mixed-effects model). You can also use the `anova.rma` function to compute $R^2$ for any two models that are known to be nested.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Higgins, J. P. T., & Thompson, S. G. (2002). Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine*, **21**, 1539–1558.

Raudenbush, S. W. (2009). Analyzing effect sizes: Random effects models. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 295–315). New York: Russell Sage Foundation.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

[rma.uni](), [rma.mh](), [rma.peto](), [rma.glmm](), [rma.mv]()

---

| profile.rma.uni | *Profile Plots for 'rma' Objects* |
|---|---|

---

## Description

Function to profile the (restricted) log-likelihood for objects of class "rma.uni" and "rma.mv".

## Usage

```
## S3 method for class 'rma.uni'
profile(fitted, xlim, ylim, steps=20, progbar=TRUE, plot=TRUE, pch=19, ...)

## S3 method for class 'rma.mv'
profile(fitted, sigma2, tau2, rho, xlim, ylim,
        steps=20, startmethod="init", progbar=TRUE, plot=TRUE, pch=19, ...)
```

## Arguments

| | |
|---|---|
| fitted | an object of class "rma.uni" or "rma.mv". |
| sigma2 | integer specifying for which $\sigma^2$ value the likelihood should be profiled (only relevant for "rma.mv" objects). |
| tau2 | integer specifying for which $\tau^2$ value the likelihood should be profiled (only relevant for "rma.mv" objects). |
| rho | integer specifying for which $\rho$ value the likelihood should be profiled (only relevant for "rma.mv" objects). |
| xlim | optional vector specifying the lower and upper limit of the parameter over which the profiling should be done. If unspecified, the function tries to set these limits automatically. |
| ylim | optional vector specifying the y-axis limits when plotting the profiled likelihood. If unspecified, the function tries to set these limits automatically. |
| steps | number of points between xlim[1] and xlim[2] (inclusive) for which the likelihood should be obtained (default is 20). |

| startmethod | method for picking starting values for the optimization. Default is "init", which starts each model fit at the default initial values. The alternative is "prev", which starts at the estimates from the previous model fit (usually faster, but also less stable for ill-defined models). |
|---|---|
| progbar | logical indicating whether a progress bar should be shown (default is TRUE). |
| plot | logical indicating whether the profile plot should be drawn after profiling is finished (default is TRUE). |
| pch | plotting symbol to use. By default, a filled circle is used. See [points] for other options. |
| ... | other arguments. |

## Details

The function fixes a particular variance component or correlation parameter of the model and then computes the maximized (restricted) log-likelihood over the remaining parameters of the model. By doing this for a range of values for the parameter that was fixed, a profile of the (restricted) log-likelihood is obtained.

For objects of class "rma.uni", the function profiles over parameter $\tau^2$. If method="ML" or method="REML" when fitting the model with the [rma.uni] function, the profiled regular/restricted log-likelihood should be maximized at the ML/REML estimate of $\tau^2$.

For objects of class "rma.mv" obtained with the [rma.mv] function, one must use the sigma2, tau2, and rho arguments to specify over which parameter the profiling should be done. Only one of these arguments can be used at a time. A single integer is used to specify the number of the parameter. Each profile plot should show a clear peak at the corresponding ML/REML estimate. If the profiled likelihood is flat (over the entire parameter space or large portions of it), then this suggests that at least some of the parameters of the model are not identifiable (and the parameter estimates obtained are essentially arbitrary).

## Value

An object of class "profile.rma.uni" or "profile.rma.mv". The object is a list containing the following components:

| sigma2 | values of $\sigma^2$ over which the likelihood was profiled (only when profiling was actually done over $\sigma^2$). |
|---|---|
| tau2 | values of $\tau^2$ over which the likelihood was profiled (only when profiling was actually done over $\tau^2$). |
| rho | values of $\rho$ over which the likelihood was profiled (only when profiling was actually done over $\rho$). |
| ll | (restricted) log-likelihood at the corresponding parameter value. |
| b | a matrix with the estimated model coefficients at the corresponding parameter value. |
| ci.lb | a matrix with the lower confidence interval bounds for the model coefficients at the corresponding parameter value. |
| ci.ub | a matrix with the upper confidence interval bounds for the model coefficients at the corresponding parameter value. |

Note that the list is returned invisibly.

**Note**

Profiling can be slow when $k$ is large and/or the model is complex (the latter applies to ″rma.mv″).

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

**References**

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

rma.uni, rma.mv

**Examples**

```
### load BCG vaccine data
data(dat.bcg)

### calculate log odds ratios and corresponding sampling variances
dat <- escalc(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model using rma.uni()
res <- rma(yi, vi, data=dat)

### profile over tau^2
profile(res)

### change data into long format
dat.long <- to.long(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### set levels of group variable ("exp" = experimental/vaccinated; "con" = control/non-vaccinated)
levels(dat.long$group) <- c("exp", "con")

### set "con" to reference level
dat.long$group <- relevel(dat.long$group, ref="con")

### calculate log odds and corresponding sampling variances
dat.long <- escalc(measure="PLO", xi=out1, mi=out2, data=dat.long)

### bivariate random-effects model using rma.mv()
res <- rma.mv(yi, vi, mods = ~ group, random = ~ group | study, struct="UN", data=dat.long)
res

### profile over tau^2_1, tau^2_2, and rho
### note: for rho, adjust region over which profiling is done ("zoom in" on area around estimate)
## Not run:
```

```
par(mfrow=c(3,1))
profile(res, tau2=1)
profile(res, tau2=2)
profile(res, rho=1, xlim=c(.90, .98))

## End(Not run)
```

---

qqnorm.rma.uni                 *Normal QQ Plots for 'rma' Objects*

---

### Description

Function to create normal QQ plots for objects of class ″rma.uni″, ″rma.mh″, and ″rma.peto″.

### Usage

```
## S3 method for class 'rma.uni'
qqnorm(y, type="rstandard", pch=19, envelope=TRUE, level=y$level,
      bonferroni=FALSE, reps=1000, smooth=TRUE, bass=0, label=FALSE, offset=0.3, ...)
## S3 method for class 'rma.mh'
qqnorm(y, type="rstandard", pch=19, label=FALSE, offset=0.3, ...)
## S3 method for class 'rma.peto'
qqnorm(y, type="rstandard", pch=19, label=FALSE, offset=0.3, ...)
## S3 method for class 'rma.glmm'
qqnorm(y, ...)
## S3 method for class 'rma.mv'
qqnorm(y, ...)
```

### Arguments

| | |
|---|---|
| y | an object of class ″rma.uni″, ″rma.mh″, or ″rma.peto″. The method is not yet implemented for objects of class ″rma.glmm″ or ″rma.mv″. |
| type | either ″rstandard″ (default) or ″rstudent″ indicating whether standardized residuals or studentized deleted residuals should be used in creating the plot. See 'Details'. |
| pch | plotting symbol to use for the observed effect sizes or outcomes. By default, a solid circle is used. See [points](#) for other options. |
| envelope | logical indicating whether a pseudo confidence envelope should be simulated and added to the plot (default is TRUE)). Only for objects of class ″rma.uni″. See 'Details'. |
| level | numerical value between 0 and 100 specifying the level of the pseudo confidence envelope (the default is to take the value from the object). |
| bonferroni | logical indicating whether the bounds of the envelope should be Bonferroni corrected. |
| reps | numerical value indicating the number of iterations to use for simulating the pseudo confidence envelope (default is 1000). |

| smooth | logical indicating whether the results from the simulation should be smoothed (default is TRUE). |
|--------|--------------------------------------------------------------------------------------------------|
| bass   | numerical value that controls the degree of smoothing (default is 0). |
| label  | argument to control the labeling of the points (default is FALSE). See 'Details'. |
| offset | argument to control the distance between the points and the corresponding labels (default is 0.3). |
| ...    | other arguments. |

### Details

The plot shows the theoretical quantiles of a normal distribution on the horizontal axis against the observed quantiles for either the standardized residuals (type="rstandard", the default) or the externally standardized residuals (type="rstudent") on the vertical axis (see residuals.rma for details on the definition of these residual types).

For reference, a line is added to the plot with slope of 1, going through the (0,0) point.

For objects of class "rma.uni", it is also possible to add a pseudo confidence envelope to the plot. The envelope is created based on the quantiles of sets of pseudo residuals simulated from the given model (for details, see Cook & Weisberg, 1982). The number of sets simulated can be controlled with the reps argument. When smooth=TRUE, the simulated bounds are smoothed with Friedman's SuperSmoother (see supsmu). The bass argument can be set to a number between 0 and 10, with higher numbers indicating increasing smoothness. If bonferroni=TRUE, the envelope bounds are Bonferroni corrected, so that the envelope can be regarded as a confidence region for all $k$ residuals simultaneously. The default however is bonferroni=FALSE, which makes the plot more sensitive to deviations from normality.

With the label argument, one can control whether points in the plot will be labeled (e.g., to identify outliers). If label="all", all points in the plot will be labeled. If label="out", points falling outside of the confidence envelope will be labeled (only available for objects of class "rma.uni"). Finally, one can also set this argument to a numeric value (between 1 and $k$), indicating how many of the most extreme points should be labeled (for example, with label=1 only the most extreme point would be labeled, while with label=3, the most extreme, and the second and third most extreme points would be labeled). With the offset argument, the distance between the labels and the corresponding points can be controlled.

### Value

A list with components:

| x | the x coordinates of the points that were plotted. |
|---|----------------------------------------------------|
| y | the y coordinates of the points that were plotted. |

Note that the list is returned invisibly.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Cook, R. D., & Weisberg, S. (1982). *Residuals and influence in regression*. London: Chapman and Hall.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Wang, M. C., & Bushman, B. J. (1998). Using the normal quantile plot to explore meta-analytic data sets. *Psychological Methods*, **3**, 46–54.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm, rma.mv

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
qqnorm(res)

### mixed-effects model with absolute latitude as a moderator
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat,
           data=dat.bcg)
qqnorm(res)
```

---

radial                    *Radial (Galbraith) Plots for 'rma' Objects*

---

## Description

Function to create radial (also called Galbraith) plots for objects of class "rma".

## Usage

```
radial(x, ...)
galbraith(x, ...)

## S3 method for class 'rma'
radial(x, center=FALSE, xlim, zlim, xlab, zlab,
       atz, aty, steps=7, level=x$level, digits=2, back="lightgray",
       transf, targs, pch=19, arc.res=100, cex, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"rma"`. |
| center | logical to indicate whether the plot should be centered horizontally at the model estimate (default is `FALSE`). |
| xlim | x-axis limits. If unspecified, the function tries to set the x-axis limits to some sensible values. |
| zlim | z-axis limits. If unspecified, the function tries to set the z-axis limits to some sensible values (note that the z-axis limits are the actual vertical limit of the plotting region). |
| xlab | title for the x-axis. If unspecified, the function tries to set an appropriate axis title. |
| zlab | title for the z-axis. If unspecified, the function tries to set an appropriate axis title. |
| atz | position for the z-axis tick marks and labels. If unspecified, these values are set by the function. |
| aty | position for the y-axis tick marks and labels. If unspecified, these values are set by the function. |
| steps | the number of tick marks for the y-axis (default is 7). Ignored when argument `aty` is used. |
| level | numerical value between 0 and 100 specifying the level of the z-axis error region (the default is to take the value from the object). |
| digits | integer specifying the number of decimal places to which the tick mark labels of the y-axis should be rounded (default is 2). |
| back | color of the z-axis error region. Set to `NA` to suppress shading of the region. |
| transf | optional argument specifying the name of a function that should be used to transform the y-axis labels (e.g., `transf=exp`). If unspecified, no transformation is used. |
| targs | optional arguments needed by the function specified via `transf`. |
| pch | plotting symbol. By default, a solid circle is used. See [points](#) for other options. |
| arc.res | integer specifying the number of line segments to use when drawing the y-axis and confidence interval arcs (default is 100). |
| cex | optional character and symbol expansion factor. If unspecified, the function tries to set this to a sensible value. |
| ... | other arguments. |

## Details

For a fixed-effects model, the plot shows the inverse of the standard errors on the horizontal axis against the individual observed effect sizes or outcomes standardized by their corresponding standard errors on the vertical axis. Since the vertical axis corresponds to standardized values, it is referred to as the z-axis within this function. On the right hand side of the plot, an arc is drawn (referred to as the y-axis within this function) corresponding to the individual observed effect sizes

or outcomes. A line projected from (0,0) through a particular point within the plot onto this arc indicates the value of the individual observed effect size or outcome for that point.

For a random-effects model, the function uses $1/\sqrt{v_i + \tau^2}$ for the horizontal axis, where $v_i$ is the sampling variance of the observed effect size or outcome and $\tau^2$ is the amount of heterogeneity as estimated based on the model. For the z-axis, $\sqrt{v_i + \tau^2}$ is used to standardize the individual observed effect sizes or outcomes.

If the model contains moderators, the function returns an error.

## Value

A data frame with components:

| | |
|---|---|
| x | the x coordinates of the points that were plotted. |
| y | the y coordinates of the points that were plotted. |
| slab | study labels of the points that were plotted. |

Note that the data frame is returned invisibly.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Galbraith, R. F. (1988). Graphical display of estimates having differing standard errors. *Technometrics*, **30**, 271–281.

Galbraith, R. F. (1988). A note on graphical presentation of estimated odds ratios from several clinical trials. *Statistics in Medicine*, **7**, 889–894.

Galbraith, R. F (1994). Some applications of radial plots. *Journal of the American Statistical Association*, **89**, 1232–1242.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm, rma.mv

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a fixed-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg, method="FE")
radial(res)

### line from (0,0) with slope equal to the log relative risk from the 4th study
```

```
abline(a=0, b=c(-1.44155119), lty="dotted")

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
radial(res)
```

---

ranktest                    *Rank Correlation Test for Funnel Plot Asymmetry for 'rma' Objects*

---

### Description

Rank correlation test for funnel plot asymmetry for objects of class "rma".

### Usage

```
ranktest(x, ...)

## S3 method for class 'rma'
ranktest(x, ...)
```

### Arguments

x               an object of class "rma".
...             other arguments.

### Details

The function carries out the rank correlation test as described by Begg and Mazumdar (1994). The test can be used to examine whether the observed outcomes and the corresponding sampling variances are correlated. A high correlation would indicate that the funnel plot is asymmetric, which may be a result of publication bias.

### Value

An object of class "ranktest.rma". The object is a list containing the following components:

tau             the estimated value of Kendall's tau rank correlation coefficient
pval            the corresponding p-value for the test that the true tau is equal to zero

The results are formated and printed with the [print.ranktest.rma](print.ranktest.rma) function.

### Note

The method does not depend on the model fitted. Therefore, regardless of the model, the results of the rank test will always be the same. See [regtest](regtest) for tests of funnel plot asymmetry that are based on regression models and model dependent.

The function makes use of the [cor.test](cor.test) function with method="kendall". If possible, an exact p-value is provided; otherwise, a large-sample approximation is used.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## References

Begg, C. B., & Mazumdar, M. (1994). Operating characteristics of a rank correlation test for publication bias. *Biometrics*, **50**, 1088–1101.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

regtest

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model
res <- rma(yi, vi, data=dat)

ranktest(res)
```

---

regtest                          *Regression Test for Funnel Plot Asymmetry for 'rma' Objects*

---

## Description

The function can be used to carry out (various versions of) Egger's regression test for funnel plot asymmetry for objects of class "rma".

## Usage

```
regtest(x, ...)

## S3 method for class 'rma'
regtest(x, model="rma", predictor="sei", ret.fit=FALSE, ...)
```

## Arguments

| | |
|---|---|
| x | an object of class `"rma"`. |
| model | either `"rma"` or `"lm"` to indicate the type of model to use for the regression test. See 'Details'. |
| predictor | either `"sei"` `"vi"`, `"ni"`, or `"ninv"` to indicate the type of independent variable to use for the regression test. See 'Details'. |
| ret.fit | logical indicating whether the full results from the fitted model should also be returned. |
| ... | other arguments. |

## Details

Various tests for funnel plot asymmetry have been suggested in the literature, including the rank correlation test by Begg and Mazumdar (1994) and the regression test by Egger et al. (1997). Extensions, modifications, and further developments of the regression test are described (among others) by Macaskill, Walter, and Irwig (2001), Sterne and Egger (2005), Harbord, Egger, and Sterne (2006), Peters et al. (2006), Rücker, Schwarzer, and Carpenter (2008), and Moreno et al. (2009). The various versions of the regression test differ in terms of the model (either a regular weighted regression with a multiplicative dispersion term or one of the meta-analytic models is used), in terms of the independent variable that the observed outcomes are hypothesized to be related to when publication bias is present (suggested predictors include the standard error, the sampling variance, the sample size, and the inverse of the sample size), and in terms of the outcome measure used (e.g., for $2 \times 2$ table data, one has the choice between various outcome measures). The idea behind the various tests is the same though: If there is a relationship between the observed outcomes and the chosen predictor, then this usually implies asymmetry in the funnel plot, which in turn may be an indication of publication bias.

The `regtest.rma` function can be used to carry out various versions of the regression test. The model is chosen via the `model` argument, with `model="lm"` for weighted regression with a multiplicative dispersion term or `model="rma"` for the meta-analytic models. In the latter case, arguments such as `method`, `weighted`, and `knha` used during the initial model fitting are also used for the regression test. Therefore, if one wants to conduct the regression test with a random/mixed-effects model, one should first fit a random-effects model with the `rma` function and then use the `regtest.rma` function on the fitted model object.

The predictor is chosen via the `predictor` argument:

- `predictor="sei"` for the standard error,
- `predictor="vi"` for the sampling variance,
- `predictor="ni"` for the sample size, and
- `predictor="ninv"` for the inverse of the sample size.

For `predictor="ni"` and `predictor="ninv"`, the object x must contain the information about the sample sizes. This will be the case when `measure` was *not* equal to `"GEN"` or the `ni` values were explicitly specified during the initial model fitting.

Finally, depending on what outcome measure was used for the model fitting will determine which outcome measure is used for the regression test.

## Value

An object of class `"regtest.rma"`. The object is a list containing the following components:

| | |
|---|---|
| `model` | the model used for the regression test. |
| `predictor` | the predictor used for the regression test. |
| `zval` | the value of the test statistic. |
| `pval` | the corresponding p-value |
| `dfs` | the degrees of freedom of the test stastistic (if the test is based on a t-distribution). |
| `fit` | the full results from the fitted model. |

The results are formated and printed with the `print.regtest.rma` function.

## Note

The classical "Egger test" is obtained by setting `model="lm"` and `predictor="sei"`. For the random/mixed-effects version of the Egger test, one should first fit a random-effects model to the data and then set `model="rma"` and `predictor="sei"` when using the `regtest.rma` function. See Sterne and Egger (2005) for details on these two types of models/tests.

All of the tests do not directly test for publication bias, but for a relationship between the observed outcomes and the chosen predictor. If such a relationship is present, then this usually implies asymmetry in the funnel plot, which in turn may be an indication of publication bias. However, it is important to keep in mind that there can be other reasons besides publication bias that could lead to asymmetry in the funnel plot.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Begg, C. B., & Mazumdar, M. (1994). Operating characteristics of a rank correlation test for publication bias. *Biometrics*, **50**, 1088–1101.

Egger, M., Davey Smith, G., Schneider, M., & Minder, C. (1997). Bias in meta-analysis detected by a simple, graphical test. *British Medical Journal*, **315**, 629–634.

Harbord, R. M., Egger, M., & Sterne, J. A. C. (2006). A modified test for small-study effects in meta-analyses of controlled trials with binary endpoints. *Statistics in Medicine*, **25**, 3443–3457.

Macaskill, P., Walter, S. D., & Irwig, L. (2001). A comparison of methods to detect publication bias in meta-analysis. *Statistics in Medicine*, **20**, 641–654.

Moreno, S. G., Sutton, A. J., Ades, A. E., Stanley, T. D., Abrams, K. R., Peters, J. L., & Cooper, N. J. (2009). Assessment of regression-based methods to adjust for publication bias through a comprehensive simulation study. *BMC Medical Research Methodology*, **9**:2.

Peters, J. L., Sutton, A. J., Jones, D. R., Abrams, K. R., & Rushton, L. (2006). Comparison of two methods to detect publication bias in meta-analysis. *Journal of the American Medical Association*, **295**, 676–680.

Rücker, G., Schwarzer, G., & Carpenter, J. (2008). Arcsine test for publication bias in meta-analyses with binary outcomes. *Statistics in Medicine*, **27**, 746–763.

Sterne, J. A. C., & Egger, M. (2005). Regression methods to detect publication and other bias in meta-analysis. In H. R. Rothstein, A. J. Sutton, & M. Borenstein (Eds.) *Publication bias in meta-analysis: Prevention, assessment, and adjustments* (pp. 99–110). Chichester, England: Wiley.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## See Also

[ranktest](#)

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### fit random-effects model
res <- rma(yi, vi, data=dat)

### classical Egger test
regtest(res, model="lm")

### random/mixed-effects version of the Egger test
regtest(res)

### more examples
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat)

regtest(res, model="lm", predictor="ni")
regtest(res, model="lm", predictor="ninv")

regtest(res, model="rma", predictor="ni")
regtest(res, model="rma", predictor="ninv")

### testing for asymmetry after accounting for the influence of a moderator
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat, data=dat)
regtest(res, model="lm")
regtest(res)
```

---

replmiss                    *Replace Missing Values in a Vector*

---

## Description

Function to replace missing (NA) values in a vector.

## Usage

```
replmiss(x, y)
```

## Arguments

| | |
|---|---|
| x | vector that may include one or more missing values. |
| y | either a scalar or a vector of the same length as x with the value(s) to replace missing values with. |

## Value

Vector x with the missing values replaced based on the scalar or vector y.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: <http://www.metafor-project.org/>
author homepage: <http://www.wvbauer.com/>

## Examples

```
x <- c(4,2,7,NA,1,NA,5)
x <- replmiss(x,0)
x

x <- c(4,2,7,NA,1,NA,5)
y <- c(2,3,6,5,8,1,2)
x <- replmiss(x,y)
x
```

---

residuals.rma            *Residual Values based on 'rma' Objects*

---

## Description

The residuals, rstandard, and rstudent functions can be used to compute residuals, corresponding standard errors, and standardized residuals for models fitted with the rma.uni, rma.mh, rma.peto, and rma.mv functions.

## Usage

```
## S3 method for class 'rma'
residuals(object, ...)

## S3 method for class 'rma.uni'
rstandard(model, digits, ...)
## S3 method for class 'rma.mh'
rstandard(model, digits, ...)
```

```
## S3 method for class 'rma.mv'
rstandard(model, digits, ...)
## S3 method for class 'rma.peto'
rstandard(model, digits, ...)

## S3 method for class 'rma.uni'
rstudent(model, digits, ...)
## S3 method for class 'rma.mh'
rstudent(model, digits, ...)
## S3 method for class 'rma.peto'
rstudent(model, digits, ...)
```

## Arguments

| | |
|---|---|
| object | an object of class "rma" (for residuals). |
| model | an object of class "rma.uni", "rma.mh", "rma.peto", or "rma.mv" (for rstandard and rstudent). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (if unspecified, the default is to take the value from the object). |
| ... | other arguments. |

## Details

The observed residuals (obtained with residuals) are simply equal to the 'observed - fitted' values.

Dividing the observed residuals by their corresponding standard errors yields (internally) standardized residuals. These can be obtained with rstandard.

The rstudent function calculates externally standardized residuals (studentized deleted residuals). The externally standardized residual for the $i$th case is obtained by deleting the $i$th case from the dataset, fitting the model based on the remaining cases, calculating the predicted value for the $i$th case based on the fitted model, taking the difference between the observed and the predicted value for the $i$th case (the deleted residual), and then standardizing the deleted residual. The standard error of the deleted residual is equal to the square root of the sampling variance of the $i$th case plus the variance of the predicted value plus the amount of (residual) heterogeneity from the fitted model (for fixed-effects models, this last part is always equal to zero).

If a particular study fits the model, its standardized residual follows (asymptotically) a standard normal distribution. A large standardized residual for a study therefore may suggest that the study does not fit the assumed model (i.e., it may be an outlier).

See also [influence.rma.uni](#) for other leave-one-out diagnostics that are useful for detecting influential cases in models fitted with the [rma.uni](#) function.

## Value

Either a vector with the observed residuals (for residuals) or an object of class "list.rma", which is a list containing the following components:

| | |
|---|---|
| resid | observed residuals (for rstandard) or deleted residuals (for rstudent). |

| | |
|---|---|
| se | corresponding standard errors. |
| z | standardized residuals (internally standardized for rstandard or externally standardized for rstudent). |

The "list.rma" object is formated and printed with print.list.rma.

## Note

Right now, the externally standardized residuals are calculated by refitting the model $k$ times. Depending on how large $k$ is, it may take a few moments to finish the calculations.

It may not be possible to fit the model after deletion of the $i$th case from the dataset. This will result in NA values for that case when calling rstudent.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. San Diego, CA: Academic Press.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., & Cheung, M. W.-L. (2010). Outlier and influence diagnostics for meta-analysis. *Research Synthesis Methods*, **1**, 112–125.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm, rma.mv, influence.rma.uni

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
rstudent(res)

### mixed-effects model with absolute latitude as a moderator
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, mods = ~ ablat,
           data=dat.bcg)
rstudent(res)
```

---

rma.glmm                    *Meta-Analysis via Generalized Linear (Mixed-Effects) Models*

---

### Description

Function to fit meta-analytic fixed- and random/mixed-effects models with or without moderators via generalized linear (mixed-effects) models. See below and the documentation of the **metafor-package** for more details on these models.

### Usage

```
rma.glmm(ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i, xi, mi, ti, ni,
         mods, measure, intercept=TRUE, data, slab, subset,
         add=1/2, to="only0", drop00=TRUE, vtype="LS",
         model="UM.FS", method="ML", tdist=FALSE,
         level=95, digits=4, btt, nAGQ=7, verbose=FALSE, control)
```

### Arguments

| | |
|---|---|
| ai | see below and the documentation of the escalc function for more details. |
| bi | see below and the documentation of the escalc function for more details. |
| ci | see below and the documentation of the escalc function for more details. |
| di | see below and the documentation of the escalc function for more details. |
| n1i | see below and the documentation of the escalc function for more details. |
| n2i | see below and the documentation of the escalc function for more details. |
| x1i | see below and the documentation of the escalc function for more details. |
| x2i | see below and the documentation of the escalc function for more details. |
| t1i | see below and the documentation of the escalc function for more details. |
| t2i | see below and the documentation of the escalc function for more details. |
| xi | see below and the documentation of the escalc function for more details. |
| mi | see below and the documentation of the escalc function for more details. |
| ti | see below and the documentation of the escalc function for more details. |
| ni | see below and the documentation of the escalc function for more details. |
| mods | optional argument to include one or more moderators in the model. A single moderator can be given as a vector of length $k$ specifying the values of the moderator. Multiple moderators are specified by giving a matrix with $k$ rows and as many columns as there are moderator variables. Alternatively, a model formula can be used to specify the model. See 'Details'. |
| measure | character string indicating the outcome measure to use for the meta-analysis. Possible options are the odds ratio ("OR"), the incidence rate ratio ("IRR"), the logit transformed proportion ("PLO"), or the log transformed incidence rate ("IRLN"). |

| | |
|---|---|
| intercept | logical indicating whether an intercept term should be added to the model (default is `TRUE`). |
| data | optional data frame containing the data supplied to the function. |
| slab | optional vector with labels for the $k$ studies. |
| subset | optional vector indicating the subset of studies that should be used for the analysis. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the observations to include. |
| add | non-negative number indicating the amount to add to zero cells, counts, or frequencies when calculating the observed outcomes of the individual studies. See below and the documentation of the [escalc](escalc) function for more details. |
| to | character string indicating when the values under add should be added (either `"only0"`, `"all"`, `"if0all"`, or `"none"`). See below and the documentation of the [escalc](escalc) function for more details. |
| drop00 | logical indicating whether studies with no cases/events (or only cases) in both groups should be dropped. See the documentation of the [escalc](escalc) function for more details. |
| vtype | character string indicating the type of sampling variances to calculate when calculating the observed outcomes. See the documentation of the [escalc](escalc) function for more details. |
| model | character string specifying the general model type to use for the analysis (either `"UM.FS"` (the default), `"UM.RS"`, `"CM.EL"`, or `"CM.AL"`). See 'Details'. |
| method | character string specifying whether a fixed- or a random/mixed-effects model should be fitted. A fixed-effects model (with or without moderators) is fitted when using `method="FE"`. Random/mixed-effects models are fitted by setting `method="ML"` (the default). See 'Details'. |
| tdist | logical specifying whether test statistics and confidence intervals should be based on the normal (when `FALSE`, the default) or the t-distribution (when `TRUE`). See 'Details'. |
| level | numerical value between 0 and 100 specifying the confidence interval level (default is 95). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (default is 4). |
| btt | optional vector of indices specifying which coefficients to include in the omnibus test of moderators. See 'Details'. |
| nAGQ | positive integer specifying the number of points per axis for evaluating the adaptive Gauss-Hermite approximation to the log-likelihood. This defaults to 7. Setting this to 1 corresponds to the Laplacian approximation. See 'Note'. |
| verbose | logical indicating whether output should be generated on the progress of the model fitting (default is `FALSE`). Can also be an integer. Values > 1 generate more verbose output. See 'Note'. |
| control | optional list of control values for the estimation algorithms. If unspecified, default values are defined inside the function. See 'Note'. |

**Details**

### Specifying the Data

The function can be used in conjunction with the following effect size or outcome measures:

- measure="OR" for odds ratios (analyzed in log units)
- measure="IRR" for incidence rate ratios (analyzed in log units)
- measure="PLO" for logit transformed proportions (i.e., log odds)
- measure="IRLN" for log transformed incidence rates.

The escalc function describes the data/arguments that should be specified/used for these measures.

### Specifying the Model

A variety of model types are available when analyzing $2 \times 2$ table data (i.e., when measure="OR") or two-group event count data (i.e., when measure="IRR"):

- model="UM.FS" for an unconditional generalized linear mixed-effects model with fixed study effects
- model="UM.RS" for an unconditional generalized linear mixed-effects model with random study effects
- model="CM.AL" for a conditional generalized linear mixed-effects model (approximate likelihood)
- model="CM.EL" for a conditional generalized linear mixed-effects model (exact likelihood).

For measure="OR", models "UM.FS" and "UM.RS" are essentially (mixed-effects) logistic regression models, while for measure="IRR", these models are (mixed-effects) Poisson regression models. A choice must be made on how to model study level variability (i.e., differences in outcomes across studies irrespective of group membership). One can choose between using fixed study effects (which means that $k$ dummy variables are added to the model) or random study effects (which means that random effects corresponding to the study factor are added to the model).

The conditional model (model="CM.EL") avoids having to model study level variability by conditioning on the total numbers of cases/events in each study. For measure="OR", this leads to a non-central hypergeometric distribution for the data within each study and the corresponding model is then a (mixed-effects) conditional logistic model. Fitting this model can be difficult and computationally expensive. When the number of cases in each study is small relative to the group sizes, one can approximate the exact likelihood by a binomial distribution, which leads to a regular (mixed-effects) logistic regression model (model="CM.AL"). For measure="IRR", the conditional model leads directly to a binomial distribution for the data within each study and the resulting model is again a (mixed-effects) logistic regression model (no approximate likelihood model is needed here).

When analyzing proportions (i.e., measure="PLO") or incidence rates (i.e., measure="IRLN") of individual groups, the model type is always a (mixed-effects) logistic or Poisson regression model, respectively (i.e., the model argument is not relevant here).

Aside from choosing the general model type, one has to decide whether to fit a fixed- or random-effects model to the data. A *fixed-effects model* is fitted by setting method="FE". A *random-effects model* is fitted by setting method="ML" (the default). Note that random-effects models with dichotomous data are often referred to as 'binomial-normal' models in the meta-analytic literature. Analogously, for event count data, such models could be referred to as 'Poisson-normal' models.

One or more moderators can be included in all of these models via the mods argument. A single moderator can be given as a (row or column) vector of length $k$ specifying the values of the moderator. Multiple moderators are specified by giving an appropriate model matrix (i.e., $\mathbf{X}$) with $k$ rows and as many columns as there are moderator variables (e.g., mods = cbind(mod1, mod2, mod3), where mod1, mod2, and mod3 correspond to the names of the variables for three moderator variables). The intercept is added to the model matrix by default unless intercept=FALSE.

Alternatively, one can use the standard [formula](#) syntax to specify the model. In this case, the mods argument should be set equal to a one-sided formula of the form mods = ~ model (e.g., mods = ~ mod1 + mod2 + mod3). Interactions, polynomial terms, and factors can be easily added to the model in this manner. When specifying a model formula via the mods argument, the intercept argument is ignored. Instead, the inclusion/exclusion of the intercept term is controlled by the specified formula (e.g., mods = ~ mod1 + mod2 + mod3 - 1 would lead to the removal of the intercept term). With moderators, a *fixed-effects with moderators model* is then fitted by setting method="FE", while a *mixed-effects model* is fitted by setting method="ML".

### Fixed-, Saturated-, and Random/Mixed-Effects Models

When fitting a particular model, actually up to three different models are fitted within the function:

- the fixed-effects model (i.e., where $\tau^2$ is set to 0),
- the saturated model (i.e., the model with a deviance of 0), and
- the random/mixed-effects model (i.e., where $\tau^2$ is estimated) (only if method="ML").

The saturated model is obtained by adding as many dummy variables to the model as needed so that the model deviance is equal to zero. Even when method="ML", the fixed-effects and saturated models are fitted, as they are used to compute the test statistics for the Wald-type and likelihood ratio tests for (residual) heterogeneity (see below).

### Omnibus Test of Parameters

For models including moderators, an omnibus test of all the model coefficients is conducted that excludes the intercept (the first coefficient) if it is included in the model. If no intercept is included in the model, then the omnibus test includes all of the coefficients in the model including the first. Alternatively, one can manually specify the indices of the coefficients to test via the btt argument. For example, with btt=c(3,4), only the third and fourth coefficient from the model would be included in the test (if an intercept is included in the model, then it corresponds to the first coefficient in the model).

### Categorical Moderators

Categorical moderator variables can be included in the model via the mods argument in the same way that appropriately (dummy) coded categorical independent variables can be included in linear models. One can either do the dummy coding manually or use a model formula together with the [factor](#) function to let R handle the coding automatically.

### Tests and Confidence Intervals

By default, the test statistics of the individual coefficients in the model (and the corresponding confidence intervals) are based on the normal distribution, while the omnibus test is based on a chi-square distribution with $m$ degrees of freedom ($m$ being the number of coefficients tested). As an alternative, one can set tdist=TRUE, which slightly mimics the Knapp and Hartung (2003) method by using a t-distribution with $k - p$ degrees of freedom for tests of individual coefficients and confidence intervals and an F-distribution with $m$ and $k - p$ degrees of freedom ($p$ being the total number of model coefficients including the intercept if it is present) for the omnibus test statistic.

**Tests for (Residual) Heterogeneity**

Two different tests for (residual) heterogeneity are automatically carried out by the function. The first is a Wald-type test, which tests the coefficients corresponding to the dummy variables added in the saturated model for significance. The second is a likelihood ratio test, which tests the same set of coefficients, but does so by computing $-2$ times the difference in the log-likelihood of the fixed-effects and the saturated model. These two tests are not identical for the types of models fitted by the rma.glmm function and may even lead to conflicting conclusions.

**Observed Outcomes of the Individual Studies**

The various models do not require the calculation of the observed outcomes of the individual studies (e.g., the observed odds ratios of the $k$ studies) and directly make use of the table/event counts. Zero cells/events are not a problem (except in extreme cases, such as when one of the two outcomes never occurs or when there are no events in any of the studies). Therefore, it is unnecessary to add some constant to the cell/event counts when there are zero cells/events.

However, for plotting and various other functions, it is necessary to calculate the observed outcomes for the $k$ studies. Here, zero cells/events can be problematic, so adding a constant value to the cell/event counts ensures that all $k$ values can be calculated. The add and to arguments are used to specify what value should be added to the cell/event counts and under what circumstances when calculating the observed outcomes. The documentation of the [escalc](escalc) function explains how the add and to arguments work. Note that drop00 is set to TRUE by default, since studies where ai=ci=0 or bi=di=0 or studies where x1i=x2i=0 are uninformative about the size of the effect.

**Value**

An object of class c("rma.glmm","rma"). The object is a list containing the following components:

| | |
|---|---|
| b | estimated coefficients of the model. |
| se | standard errors of the coefficients. |
| zval | test statistics of the coefficients. |
| pval | p-values for the test statistics. |
| ci.lb | lower bound of the confidence intervals for the coefficients. |
| ci.ub | upper bound of the confidence intervals for the coefficients. |
| vb | variance-covariance matrix of the estimated coefficients. |
| tau2 | estimated amount of (residual) heterogeneity. Always 0 when method="FE". |
| sigma2 | estimated amount of study level variability (only for model="UM.RS"). |
| k | number of studies included in the model. |
| p | number of coefficients in the model (including the intercept). |
| m | number of coefficients included in the omnibus test of coefficients. |
| QE.Wld | Wald-type test statistic for the test of (residual) heterogeneity. |
| QEp.Wld | p-value for the Wald-type test of (residual) heterogeneity. |
| QE.LRT | likelihood ratio test statistic for the test of (residual) heterogeneity. |
| QEp.LRT | p-value for the likelihood ratio test of (residual) heterogeneity. |
| QM | test statistic for the omnibus test of coefficients. |

| QMp | p-value for the omnibus test of coefficients. |
| I2 | value of $I^2$. |
| H2 | value of $H^2$. |
| int.only | logical that indicates whether the model is an intercept-only model. |
| yi, vi, X | the vector of outcomes, the corresponding sampling variances, and the model matrix. |
| fit.stats | a list with the log-likelihood, deviance, AIC, BIC, and AICc values. |
| ... | some additional elements/values. |

The results of the fitted model are formated and printed with the `print.rma.glmm` function. If fit statistics should also be given, use `summary.rma` (or use the `fitstats.rma` function to extract them).

## Note

Fitting the various types of models requires several different iterative algorithms:

- For model="UM.FS" and model="CM.AL", iteratively reweighted least squares (IWLS) as implemented in the `glm` function is used for fitting the fixed-effects and the saturated models. For method="ML", adaptive Gauss-Hermite quadrature as implemented in the `glmer` function is used. The same applies when model="CM.EL" is used in combination with measure="IRR" or when measure="PLO" or measure="IRLN" (regardless of the model type).

- For model="UM.RS", adaptive Gauss-Hermite quadrature as implemented in the `glmer` function is used to fit all of the models.

- For model="CM.EL" and measure="OR", the quasi-Newton method ("BFGS") as implemented in the `optim` function is used by default for fitting the fixed-effects and the saturated models. For method="ML", the same algorithm is used, together with adaptive quadrature as implemented in the `integrate` function (for the integration over the density of the non-central hypergeometric distribution).

When model="CM.EL" and measure="OR", actually model="CM.AL" is used first to obtain starting values for `optim`, so either 4 (if method="FE") or 6 (if method="ML") models need to be fitted in total.

Various control parameters can be adjusted via the control argument:

- optimizer is set by default to `"optim"`, but can be set to `"nlminb"` or one of the optimizers from the **minqa** package (i.e., `"bobyqa"`, `"newuoa"`, or `"uobyqa"`),

- optmethod is used to set the method argument for `optim` (default is "BFGS"),

- optCtrl is a list of named arguments to be passed on to the control argument of the chosen optimizer,

- glmCtrl is a list of named arguments to be passed on to the control argument of the `glm` function,

- glmerCtrl is a list of named arguments to be passed on to the control argument of the `glmer` function, and

- intCtrl is a list of named arguments (i.e., rel.tol and subdivisions) to be passed on to the `integrate` function.

Also, for `glmer`, the nAGQ argument is used to specify the number of quadrature points. The default value is 7, which should provide sufficient accuracy in the evaluation of the log-likelihood in most cases, but at the expense of speed. Setting this to 1 corresponds to the Laplacian approximation (which is faster, but less accurate).

Information on the progress of the various algorithms is obtained by setting verbose=TRUE or with control=list(verbose=TRUE). Since fitting the various models can be computationally expensive, this option is quite useful to determine how the model fitting is progressing.

For model="CM.EL" and measure="OR", optimization involves repeated calculation of the density of the non-central hypergeometric distribution. When method="ML", this also requires integration over the same density. This is currently implemented in a rather brute-force manner and may not be numerically stable, especially when models with moderators are fitted. Stability can be improved by scaling the moderators in a similar manner (i.e., don't use a moderator that is coded 0 and 1, while another uses values in the 1000s). For models with an intercept and moderators, the function actually rescales (non-dummy) variables to z-scores during the model fitting (results are given after back-scaling, so this should be transparent to the user). For models without an intercept, this is not done, so sensitivity analyses are highly recommended here (to ensure that the results do not depend on the scaling of the moderators).

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Agresti, A. (2002). *Categorical data analysis* (2nd. ed). Hoboken, NJ: Wiley.

Bagos, P. G., & Nikolopoulos, G. K. (2009). Mixed-effects Poisson regression models for meta-analysis of follow-up studies with constant or varying durations. *The International Journal of Biostatistics*, **5**(1), article 21.

van Houwelingen, H. C., Zwinderman, K. H., & Stijnen, T. (1993). A bivariate approach to meta-analysis. *Statistics in Medicine*, **12**, 2273–2284.

Stijnen, T., Hamza, T. H., & Ozdemir, P. (2010). Random effects meta-analysis of event outcome in the framework of the generalized linear mixed model with applications in sparse data. *Statistics in Medicine*, **29**, 3046–3067.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

### See Also

rma.uni, rma.mh, rma.peto, and rma.mv for other model fitting functions.

dat.nielweise2007, dat.nielweise2008, and dat.collins1985a for further examples of the use of the rma.glmm function.

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### random-effects model using rma.uni() (standard RE model analysis)
rma(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
    data=dat.bcg, method="ML", verbose=TRUE)

### random-effects models using rma.glmm() (require 'lme4' package)

### unconditional model with fixed study effects
## Not run:
rma.glmm(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
         data=dat.bcg, model="UM.FS", verbose=TRUE)

## End(Not run)

### unconditional model with random study effects
## Not run:
rma.glmm(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
         data=dat.bcg, model="UM.RS", verbose=TRUE)

## End(Not run)

### conditional model with approximate likelihood
## Not run:
rma.glmm(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
         data=dat.bcg, model="CM.AL", verbose=TRUE)

## End(Not run)

### conditional model with exact likelihood (be patient!)
### note: fitting this model is very slow, so be patient
## Not run:
rma.glmm(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
         data=dat.bcg, model="CM.EL", verbose=TRUE)

## End(Not run)
```

---

rma.mh                          *Meta-Analysis via the Mantel-Haenszel Method*

---

### Description

Function to fit fixed-effects models to $2 \times 2$ table and person-time data via the Mantel-Haenszel method. See below and the documentation of the **metafor-package** for more details on these models.

## Usage

```
rma.mh(ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i,
       measure="OR", data, slab, subset,
       add=1/2, to="only0", drop00=TRUE,
       correct=TRUE, level=95, digits=4, verbose=FALSE)
```

## Arguments

| | |
|---|---|
| ai | vector to specify the $2 \times 2$ table frequencies (upper left cell). See below and the documentation of the [escalc](#) function for more details. |
| bi | vector to specify the $2 \times 2$ table frequencies (upper right cell). See below and the documentation of the [escalc](#) function for more details. |
| ci | vector to specify the $2 \times 2$ table frequencies (lower left cell). See below and the documentation of the [escalc](#) function for more details. |
| di | vector to specify the $2 \times 2$ table frequencies (lower right cell). See below and the documentation of the [escalc](#) function for more details. |
| n1i | vector to specify the group sizes or row totals (first group). See below and the documentation of the [escalc](#) function for more details. |
| n2i | vector to specify the group sizes or row totals (second group). See below and the documentation of the [escalc](#) function for more details. |
| x1i | vector to specify the number of events (first group). See below and the documentation of the [escalc](#) function for more details. |
| x2i | vector to specify the number of events (second group). See below and the documentation of the [escalc](#) function for more details. |
| t1i | vector to specify the total person-times (first group). See below and the documentation of the [escalc](#) function for more details. |
| t2i | vector to specify the total person-times (second group). See below and the documentation of the [escalc](#) function for more details. |
| measure | character string indicating the outcome measure to use for the meta-analysis. Possible options are the odds ratio (″OR″), the relative risk (″RR″), the risk difference (″RD″), the incidence rate ratio (″IRR″), or the incidence rate difference (″IRD″). |
| data | optional data frame containing the data supplied to the function. |
| slab | optional vector with labels for the $k$ studies. |
| subset | optional vector indicating the subset of tables that should be used for the analysis. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the tables to include. |
| add | non-negative number indicating the amount to add to zero cells, counts, or frequencies when calculating the observed outcomes of the individual studies. Can also be a vector of two numbers, where the first number is used in the calculation of the observed outcomes and the second number is used when applying the Mantel-Haenszel method. See below and the documentation of the [escalc](#) function for more details. |

| | |
|---|---|
| to | character string indicating when the values under add should be added (either "only0", "all", "if0all", or "none"). Can also be a character vector, where the first string again applies when calculating the observed outcomes and the second string when applying the Mantel-Haenszel method. See below and the documentation of the [escalc](#) function for more details. |
| drop00 | logical indicating whether studies with no cases/events (or only cases) in both groups should be dropped when calculating the observed outcomes (the outcomes for such studies are set to NA). Can also be a vector of two logicals, where the first applies to the calculation of the observed outcomes and the second when applying the Mantel-Haenszel method. See below and the documentation of the [escalc](#) function for more details. |
| correct | logical indicating whether to apply a continuity correction when computing the Cochran-Mantel-Haenszel test statistic. |
| level | numerical value between 0 and 100 specifying the confidence interval level (default is 95). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (default is 4). |
| verbose | logical indicating whether output should be generated on the progress of the model fitting (default is FALSE). |

### Details

#### Specifying the Data

When the outcome measure is either the odds ratio (measure="OR"), relative risk (measure="RR"), or risk difference (measure="RD"), the studies are assumed to provide data in terms of $2 \times 2$ tables of the form:

| | outcome 1 | outcome 2 | total |
|---|---|---|---|
| group 1 | ai | bi | n1i |
| group 2 | ci | di | n2i |

where ai, bi, ci, and di denote the cell frequencies and n1i and n2i the row totals. For example, in a set of randomized clinical trials (RCTs) or cohort studies, group 1 and group 2 may refer to the treatment (exposed) and placebo/control (not exposed) group, with outcome 1 denoting some event of interest (e.g., death) and outcome 2 its complement. In a set of case-control studies, group 1 and group 2 may refer to the group of cases and the group of controls, with outcome 1 denoting, for example, exposure to some risk factor and outcome 2 non-exposure. For these outcome measures, one needs to specify either ai, bi, ci, and di or alternatively ai, ci, n1i, and n2i.

Alternatively, when the outcome measure is the incidence rate ratio (measure="IRR") or the incidence rate difference (measure="IRD"), the studies are assumed to provide data in terms of tables of the form:

| | events | person-time |
|---|---|---|
| group 1 | x1i | t1i |
| group 2 | x2i | t2i |

where x1i and x2i denote the number of events in the first and the second group, respectively, and t1i and t2i the corresponding total person-times at risk.

### Mantel-Haenszel Method

An approach for aggregating table data of these types was suggested by Mantel and Haenszel (1959) and later extended by various authors (see references). The Mantel-Haenszel method provides a weighted estimate under a fixed-effects model. The method is particularly advantageous when aggregating a large number of studies with small sample sizes (the so-called sparse data or increasing strata case).

When analyzing odds ratios, the Cochran-Mantel-Haenszel (CMH) test (Cochran, 1954; Mantel & Haenszel, 1959) and Tarone's test for heterogeneity (Tarone, 1985) are also provided (by default, the CMH test statistic is computed with the continuity correction; this can be switched off with correct=FALSE). When analyzing incidence rate ratios, the Mantel-Haenszel (MH) test (Rothman et al., 2008) for person-time data is also provided (again, the correct argument controls whether the continuity correction is applied). When analyzing odds ratios, relative risks, or incidence rate ratios, the printed results are given both in terms of the log and the raw units (for easier interpretation).

### Observed Outcomes of the Individual Studies

The Mantel-Haenszel method itself does not require the calculation of the observed outcomes of the individual studies (e.g., the observed odds or incidence rate ratios of the $k$ studies) and directly makes use of the table/event counts. Zero cells/events are not a problem (except in extreme cases, such as when one of the two outcomes never occurs in any of the $2 \times 2$ tables or when there are no events for one of the two groups in any of the tables). Therefore, it is unnecessary to add some constant to the cell/event counts when there are zero cells/events.

However, for plotting and various other functions, it is necessary to calculate the observed outcomes for the $k$ studies. Here, zero cells/events can be problematic, so adding a constant value to the cell/event counts ensures that all $k$ values can be calculated. The add and to arguments are used to specify what value should be added to the cell/event counts and under what circumstances when calculating the observed outcomes and when applying the Mantel-Haenszel method. Similarly, the drop00 argument is used to specify how studies with no cases/events (or only cases) in both groups should be handled. The documentation of the [escalc](#) function explains how the add, to, and drop00 arguments work. If only a single value for these arguments is specified (as per default), then these values are used when calculating the observed outcomes and no adjustment to the cell/event counts is made when applying the Mantel-Haenszel method. Alternatively, when specifying two values for these arguments, the first value applies when calculating the observed outcomes and the second value when applying the Mantel-Haenszel method.

Note that drop00 is set to TRUE by default. Therefore, the observed outcomes for studies where ai=ci=0 or bi=di=0 or studies where x1i=x2i=0 are set to NA. When applying the Mantel-Haenszel method, such studies are not explicitly dropped (unless the second value of drop00 argument is also set to TRUE), but this is practically not necessary, as they do not actually influence the results (assuming no adjustment to the cell/event counts are made when applying the Mantel-Haenszel method).

### Value

An object of class c("rma.mh","rma"). The object is a list containing the following components:

b                 aggregated log odds ratio, log relative risk, risk difference, log rate ratio, or rate difference.

| se | standard error of the aggregated value. |
| zval | test statistics of the aggregated value. |
| pval | p-value for the test statistic. |
| ci.lb | lower bound of the confidence interval. |
| ci.ub | upper bound of the confidence interval. |
| QE | test statistic for the test of heterogeneity. |
| QEp | p-value for the test of heterogeneity. |
| MH | Cochran-Mantel-Haenszel test statistic (measure="OR") or Mantel-Haenszel test statistic (measure="IRR"). |
| MHp | corresponding p-value. |
| TA | Tarone's heterogeneity test statistic (only when measure="OR"). |
| TAp | corresponding p-value (only when measure="OR"). |
| k | number of tables included in the analysis. |
| yi, vi | the vector of outcomes and corresponding sampling variances. |
| fit.stats | a list with the log-likelihood, deviance, AIC, BIC, and AICc values under the unrestricted and restricted likelihood. |
| ... | some additional elements/values. |

The results of the fitted model are formated and printed with the print.rma.mh function. If fit statistics should also be given, use summary.rma (or use the fitstats.rma function to extract them).

The residuals.rma, rstandard.rma.mh, and rstudent.rma.mh functions extract raw and standardized residuals. Leave-one-out diagnostics can be obtained with leave1out.rma.mh.

Forest, funnel, radial, L'Abbé, and Baujat plots can be obtained with forest.rma, funnel.rma, radial.rma, labbe.rma, and baujat.rma.mh. The qqnorm.rma.mh function provides normal QQ plots of the standardized residuals. One can also just call plot.rma.mh on the fitted model object to obtain various plots at once.

A cumulative meta-analysis (i.e., adding one obervation at a time) can be obtained with cumul.rma.mh.

Other extractor functions include coef.rma, vcov.rma, logLik.rma, deviance.rma, AIC.rma, and BIC.rma.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Cochran, W. G. (1954). Some methods for strengthening the common $\chi^2$ tests. *Biometrics*, **10**, 417–451.

Greenland, S., & Robins, J. M. (1985). Estimation of a common effect parameter from sparse follow-up data. *Biometrics*, **41**, 55–68.

Mantel, N., & Haenszel, W. (1959). Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute*, **22**, 719–748.

Nurminen, M. (1981). Asymptotic efficiency of general noniterative estimators of common relative risk. *Biometrika*, **68**, 525–530.

Robins, J., Breslow, N., & Greenland, S. (1986). Estimators of the Mantel-Haenszel variance consistent in both sparse data and large-strata limiting models. *Biometrics*, **42**, 311–323.

Rothman, K. J., Greenland, S., & Lash, T. L. (2008). *Modern epidemiology* (3rd ed.). Philadelphia: Lippincott Williams & Wilkins.

Sato, T., Greenland, S., & Robins, J. M. (1989). On the variance estimator for the Mantel-Haenszel risk difference. *Biometrics*, **45**, 1323–1324.

Tarone, R. E. (1981). On summary estimators of relative risk. *Journal of Chronic Diseases*, **34**, 463–468.

Tarone, R. E. (1985). On heterogeneity tests based on efficient scores. *Biometrika*, **72**, 91–95.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.glmm, rma.peto, and rma.mv for other model fitting functions.

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the (log) odds ratios using the Mantel-Haenszel method
rma.mh(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### meta-analysis of the (log) relative risks using the Mantel-Haenszel method
rma.mh(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
```

---

rma.mv                        *Meta-Analysis via Multivariate/Multilevel Linear (Mixed-Effects) Models*

---

## Description

Function to fit meta-analytic multivariate/multilevel fixed- and random/mixed-effects models with or without moderators via linear (mixed-effects) models. See below and the documentation of the **metafor-package** for more details on these models.

## Usage

```
rma.mv(yi, V, W, mods, random, struct="CS", intercept=TRUE,
       data, slab, subset, method="REML", tdist=FALSE,
       level=95, digits=4, btt, R, Rscale="cor",
       sigma2, tau2, rho, sparse=FALSE, verbose=FALSE, control)
```

## Arguments

| | |
|---|---|
| yi | vector of length $k$ with the observed effect sizes or outcomes. See 'Details'. |
| V | vector of length $k$ with the corresponding sampling variances or a $k \times k$ variance-covariance matrix of the sampling errors. See 'Details'. |
| W | optional argument to specify a vector of length $k$ with user-defined weights or a $k \times k$ user-defined weight matrix. See 'Details'. |
| mods | optional argument to include one or more moderators in the model. A single moderator can be given as a vector of length $k$ specifying the values of the moderator. Multiple moderators are specified by giving a matrix with $k$ rows and as many columns as there are moderator variables. Alternatively, a model [formula](formula) can be used to specify the model. See 'Details'. |
| random | either a single one-sided formula or list of one-sided formulas to specify the random-effects structure of the model. See 'Details'. |
| struct | character string to specify the variance structure of an ~ inner \| outer formula in the random argument. Either "CS" for compound symmetry, "HCS" for heteroscedastic compound symmetry, "UN" for an unstructured variance-covariance matrix, "AR" for an AR(1) autoregressive structure, or "HAR" for a heteroscedastic AR(1) autoregressive structure. See 'Details'. |
| intercept | logical indicating whether an intercept term should be added to the model (default is TRUE). |
| data | optional data frame containing the data supplied to the function. |
| slab | optional vector with labels for the $k$ outcomes/studies. |
| subset | optional vector indicating the subset of outcomes/studies that should be used for the analysis. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the observations to include. |
| method | character string specifying whether the model should be fitted via maximum-likelihood ("ML") or via restricted maximum-likelihood ("REML") estimation. Default is "REML". |
| tdist | logical specifying whether test statistics and confidence intervals should be based on the normal (when FALSE, the default) or the t-distribution (when TRUE). See 'Details'. |
| level | numerical value between 0 and 100 specifying the confidence interval level (default is 95). |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (default is 4). |
| btt | optional vector of indices specifying which coefficients to include in the omnibus test of moderators. See 'Details'. |
| R | an optional named list of known correlation matrices corresponding to (some of) the components specified via the random argument. See 'Details'. |
| Rscale | character string, integer, or logical indicating how matrices specified via the R argument should be scaled. See 'Details'. |

sigma2            optional numerical vector (of the same length as the number of random inter-
                  cept components specified via the random argument) to fix the corresponding
                  $\sigma^2$ value(s). A specific $\sigma^2$ value can be fixed by setting the corresponding ele-
                  ment of this argument to the desired value. A specific $\sigma^2$ value will be estimated
                  if the corresponding element is set equal to NA. See 'Details'.

tau2              optional numerical value (for struct="CS" or struct="AR") or vector (for
                  struct="HCS", struct="UN", or struct="HAR") to fix the amount of (residual)
                  heterogeneity in the levels of the inner factor corresponding to an ~ inner | outer
                  formula specified in the random argument. A numeric value fixes a particular $\tau^2$
                  value, while NA means that the value should be estimated. See 'Details'.

rho               optional numerical value (for struct="CS", struct="HCS", struct="AR", or
                  struct="HAR") or vector (for struct="UN") to fix the correlation between lev-
                  els of the inner factor corresponding to an ~ inner | outer formula specified
                  in the random argument. A numeric value fixes a particular $\rho$ value, while NA
                  means that the value should be estimated. See 'Details'.

sparse            logical indicating whether the function should use sparse matrix objects to the
                  extent possible (can speed up model fitting substantially for certain models). See
                  'Note'.

verbose           logical indicating whether output should be generated on the progress of the
                  model fitting (default is FALSE). Can also be an integer. Values > 1 generate
                  more verbose output. See 'Note'.

control           optional list of control values for the estimation algorithms. If unspecified, de-
                  fault values are defined inside the function. See 'Note'.

### Details

#### Specifying the Data

The function can be used in conjunction with any of the usual effect size or outcome measures
used in meta-analyses (e.g., log odds ratios, log relative risks, risk differences, mean differences,
standardized mean differences, raw correlation coefficients, correlation coefficients transformed
with Fisher's r-to-z transformation, and so on). Simply specify the observed outcomes via the yi
argument and the corresponding sampling variances via the V argument. In case the sampling errors
are correlated, then one can specify the entire variance-covariance matrix of the sampling errors via
the V argument.

The [escalc](#) function can be used to compute a wide variety of effect size and outcome measures
(and the corresponding sampling variances) based on summary statistics. Equations for computing
the covariance between sampling errors for a variety of different effect size or outcome measures
can be found, for example, in Gleser and Olkin (2009). For raw and Fisher's r-to-z transformed
correlations, one can find suitable equations, for example, in Steiger (1980).

#### Specifying Fixed Effects

With rma.mv(yi, V), a fixed-effects model is fitted to the data (note: arguments struct, sigma2,
tau2, rho, R, and Rscale are not relevant then and are ignored). The model is then simply given
by $\mathbf{y} \sim N(\mathbf{1}\beta_0, \mathbf{V})$, where $\mathbf{y}$ is the (column) vector with the observed effect sizes or outcomes, $\mathbf{1}$
is a column vector of 1's, $\beta_0$ is the (average) true effect size or outcome, and $\mathbf{V}$ is the variance-
covariance matrix of the sampling errors (if a vector of sampling variances is provided via the V
argument, then $\mathbf{V}$ is assumed to be diagonal).

One or more moderators can be included in the model via the mods argument. A single moderator can be given as a (row or column) vector of length $k$ specifying the values of the moderator. Multiple moderators are specified by giving an appropriate model matrix (i.e., $\mathbf{X}$) with $k$ rows and as many columns as there are moderator variables (e.g., mods = cbind(mod1, mod2, mod3), where mod1, mod2, and mod3 correspond to the names of the variables for the three moderator variables). The intercept is added to the model matrix by default unless intercept=FALSE.

Alternatively, one can use the standard [formula](#) syntax to specify the model. In this case, the mods argument should be set equal to a one-sided formula of the form mods = ~ model (e.g., mods = ~ mod1 + mod2 + mod3). Interactions, polynomial terms, and factors can be easily added to the model in this manner. When specifying a model formula via the mods argument, the intercept argument is ignored. Instead, the inclusion/exclusion of the intercept term is controlled by the specified formula (e.g., mods = ~ mod1 + mod2 + mod3 - 1 would lead to the removal of the intercept term). One can also directly specify moderators via the yi argument (e.g., rma.mv(yi ~ mod1 + mod2 + mod3, V)). In that case, the mods argument is ignored and the inclusion/exclusion of the intercept term again is controlled by the specified formula.

With moderators included, the model is then given by $\mathbf{y} \sim N(\mathbf{X}\beta, \mathbf{V})$, where $\mathbf{X}$ denotes the model matrix containing the moderator values (and possibly the intercept) and $\beta$ is a column vector containing the corresponding model coefficients. The model coefficients (i.e., $\beta$) are then estimated with $\mathbf{b} = (\mathbf{X}'\mathbf{W}\mathbf{X}')^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}$, where $\mathbf{W} = \mathbf{V}^{-1}$. With the W argument, one can also specify user-defined weights (or a weight matrix).

**Specifying Random Effects**

One can fit random/mixed-effects models to the data by specifying the desired random effects structure via the random argument. The random argument is either a single one-sided formula or a list of one-sided formulas. A formula specified via this argument can be of the form random = ~ 1 | id. Such a formula adds random effects corresponding to the grouping variable/factor id to the model. Effects or outcomes with the same value/level of the id variable/factor receive the same random effect, while effects or outcomes with different values/levels of the id variable/factor are assumed to be independent. The variance component corresponding to such a formula is denoted by $\sigma^2$. An arbitrary number of such formulas can be specified as a list of formulas (e.g., random = list(~ 1 | id1, ~ 1 | id2)), with variance components $\sigma_1^2$, $\sigma_2^2$, and so on.

Such random effects components are useful to model clustering (i.e., correlation) induced by a multilevel structure in the data (e.g., effects derived from the same paper, lab, research group, or species may be more similar to each other than effects derived from different papers, labs, research groups, or species). See, for example, Konstantopoulos (2011) and Nakagawa and Santos (2012) for more details.

See [dat.konstantopoulos2011](#) for an example analysis with multilevel meta-analytic data.

In addition or alternatively to specifying one or multiple ~ 1 | id terms, the random argument can also contain one (and only one!) formula of the form ~ inner | outer. Effects or outcomes with different values/levels of the outer grouping variable/factor are assumed to be independent, while effects or outcomes with the same value/level of the outer grouping variable/factor share correlated random effects corresponding to the levels of the inner grouping variable/factor (note that the inner grouping variable must be either a factor or a character variable). The struct argument is used to specify the variance structure corresponding to the inner variable/factor. With struct="CS", a compound symmetric structure is assumed (i.e., a single variance component $\tau^2$ corresponding to all values/levels of the inner variable/factor and a single correlation coefficient $\rho$ for the correlation between different values/levels). With struct="HCS", a heteroscedastic compound symmetric structure is assumed (with variance components $\tau_1^2$, $\tau_2^2$, and so on, corresponding

to the values/levels of the `inner` variable/factor and a single correlation coefficient $\rho$ for the correlation between different values/levels). With struct="UN", an unstructured variance-covariance matrix is assumed (with variance components $\tau_1^2$, $\tau_2^2$, and so on, corresponding to the values/levels of the `inner` variable/factor and correlation coefficients $\rho_{12}$, $\rho_{13}$, $\rho_{23}$, and so on, for the various combinations of the values/levels of the `inner` variable/factor). For example, for an `inner` grouping variable/factor with four levels, the three structures correspond to variance-covariance matrices of the form:

struct="CS"

$$\begin{bmatrix} \tau^2 & \rho\tau^2 & \rho\tau^2 & \rho\tau^2 \\ & \tau^2 & \rho\tau^2 & \rho\tau^2 \\ & & \tau^2 & \rho\tau^2 \\ & & & \tau^2 \end{bmatrix}$$

struct="HCS"

$$\begin{bmatrix} \tau_1^2 & \rho\tau_1\tau_2 & \rho\tau_1\tau_3 & \rho\tau_1\tau_4 \\ & \tau_2^2 & \rho\tau_2\tau_3 & \rho\tau_2\tau_4 \\ & & \tau_3^2 & \rho\tau_3\tau_4 \\ & & & \tau_4^2 \end{bmatrix}$$

struct="UN"

$$\begin{bmatrix} \tau_1^2 & \rho_{12}\tau_1\tau_2 & \rho_{13}\tau_1\tau_3 & \rho_{14}\tau_1\tau_4 \\ & \tau_2^2 & \rho_{23}\tau_2\tau_3 & \rho_{24}\tau_2\tau_4 \\ & & \tau_3^2 & \rho_{34}\tau_4\tau_4 \\ & & & \tau_4^2 \end{bmatrix}$$

With the `outer` factor corresponding to a study id variable and the `inner` factor corresponding to a variable indicating the treatment type or study arm, such a random effects component could be used to estimate how strongly different true effects or outcomes within the same study are correlated and/or whether the amount of heterogeneity differs across different treatment types/arms. Network meta-analyses (also called mixed treatment comparison meta-analyses) will typically require such a random effects component (e.g., Salanti et al., 2008). The meta-analytic bivariate model (e.g., van Houwelingen, Arends, & Stijnen, 2002) can also be fitted in this manner (see the examples below). The `inner` factor could also correspond to a variable indicating the outcome variable, which allows for fitting multivariate models when there are multiple correlated effects per study (e.g., Berkey et al., 1998; Kalaian & Raudenbush, 1996).

See `dat.berkey1998` for an example of a multivariate meta-analysis with multiple outcomes. See `dat.hasselblad1998` for an example of a network meta-analysis.

For meta-analyses of studies reporting outcomes at multiple time points, it may also be reasonable to assume that the true effects are correlated according to an autoregressive structure over time (Ishak et al., 2007; Trikalinos & Olkin, 2012). For this purpose, one can also choose struct="AR", corresponding to a structure with a single variance component $\tau^2$ and AR(1) autocorrelation among the random effects. The values of the `inner` variable/factor should then reflect the various time points, with increasing values reflecting later time points. Finally, struct="HAR" allows for fitting a heteroscedastic AR(1) structure (with variance components $\tau_1^2$, $\tau_2^2$, and so on). For example, for an `inner` grouping variable/factor with four levels (i.e., time points), these structures correspond to variance-covariance matrices of the form:

struct="AR"

$$\begin{bmatrix} \tau^2 & \rho\tau^2 & \rho^2\tau^2 & \rho^3\tau^2 \\ & \tau^2 & \rho\tau^2 & \rho^2\tau^2 \\ & & \tau^2 & \rho\tau^2 \\ & & & \tau^2 \end{bmatrix}$$

struct="HAR"

$$\begin{bmatrix} \tau_1^2 & \rho\tau_1\tau_2 & \rho^2\tau_1\tau_3 & \rho^3\tau_1\tau_4 \\ & \tau_2^2 & \rho\tau_2\tau_3 & \rho^2\tau_2\tau_4 \\ & & \tau_3^2 & \rho\tau_3\tau_4 \\ & & & \tau_4^2 \end{bmatrix}$$

See `dat.fine1993` and `dat.ishak2007` for examples involving such structures.

When the `random` argument contains a formula of the form ~ 1 | id, one can use the (optional) argument R to specify a corresponding known correlation matrix of the random effects (i.e., R = list(id = Cor), where Cor is the correlation matrix). In that case, effects or outcomes with the same value/level of the id variable/factor receive the same random effect, while effects or outcomes with different values/levels of the id variable/factor receive random effects that are correlated as specified in the corresponding correlation matrix given via the R argument. The column/row names of the correlation matrix given via the R argument must therefore contain all of the

values/levels of the id variable/factor. When the random argument contains multiple formulas of the form ~ 1 | id, one can specify known correlation matrices for none, some, or all of those terms (e.g., with random = list(~ 1 | id1, ~ 1 | id2), one could specify R = list(id1 = Cor1) or R = list(id1 = Cor1, id2 = Cor2), where Cor1 and Cor2 are the correlation matrices corresponding to the grouping variables/factors id1 and id2, respectively).

Random effects with a known (or at least approximately known) correlation structure are useful in a variety of contexts. For example, such components can be used to account for the correlations induced by a shared phylogenetic history among organisms (e.g., plants, fungi, animals). In that case, ~ 1 | species is used to specify the species and argument R is used to specify the phylogenetic correlation matrix of the species studied in the meta-analysis. The corresponding variance component then indicates how much variance/heterogeneity is attributable to the specified phylogeny. See Nakagawa and Santos (2012) for more details. As another example, in a genetic meta-analysis studying disease association for several single nucleotide polymorphisms (SNPs), linkage disequilibrium (LD) among the SNPs can induce an approximately known degree of correlation among the effects. In that case, ~ 1 | snp could be used to specify the SNP and R the corresponding LD correlation map for the SNPs included in the meta-analysis.

The Rscale argument controls how matrices specified via the R argument are scaled. With Rscale="none" (or Rscale=0 or Rscale=FALSE), no scaling is used. With Rscale="cor" (or Rscale=1 or Rscale=TRUE), the [cov2cor](#) function is used to ensure that the matrices are correlation matrices (assuming they were covariance matrices to begin with). With Rscale="cor0" (or Rscale=2), first [cov2cor](#) is used and then the elements of each correlation matrix are scaled with $(R - min(R))/(1 - min(R))$ (this ensures that a correlation of zero in a phylogenetic correlation matrix corresponds to the split at the root node of the tree comprising the species that are actually analyzed). Finally, Rscale="cov0" (or Rscale=3) only rescales with $(R - min(R))$ (which ensures that a phylogenetic covariance matrix is rooted at the lowest split).

Together with the variance-covariance matrix of the sampling errors (i.e., $\mathbf{V}$), the random effects structure of the model implies a marginal variance-covariance matrix of the observed outcomes. Once estimates of the variance components (i.e., $\sigma^2$, $\tau^2$, and/or $\rho$ values) have been obtained, the estimated marginal variance-covariance matrix can be constructed (denoted by $\mathbf{M}$). The model coefficients (i.e., $\beta$) are then estimated with $\mathbf{b} = (\mathbf{X'WX'})^{-1}\mathbf{X'Wy}$, where $\mathbf{W} = \mathbf{M}^{-1}$. With the W argument, one can again specify user-defined weights (or a weight matrix).

**Fixing Variance Components and/or Correlations**

Arguments sigma2, tau2, and rho can be used to fix particular variance components and/or correlations at a given value. This is useful for sensitivity analyses (e.g., for plotting the regular/restricted log-likelihood as a function of a particular variance component or correlation) or for imposing a desired variance-covariance structure on the data.

For example, if random = list(~ 1 | id1, ~ 1 | id2), then sigma2 must be of length 2 (corresponding to $\sigma_1^2$ and $\sigma_2^2$) and a fixed value can be assigned to either or both variance components. Setting a particular component to NA means that the component will be estimated by the function.

Argument tau2 is only relevant when the random argument contains an ~ inner | outer formula. In that case, if the tau2 argument is used, it must be either of length 1 (for struct="CS" and struct="AR") or of the same length as the number of levels of the inner factor (for struct="HCS", struct="UN", or struct="HAR"). A numeric value in the tau2 argument then fixes the corresponding variance component to that value, while NA means that the component will be estimated. Similarly, if argument rho is used, it must be either of length 1 (for struct="CS", struct="HCS", struct="AR", or struct="HAR") or of length $lvls(lvls - 1)/2$ (for struct="UN"), where $lvls$

denotes the number of levels of the inner factor. Again, a numeric value fixes the corresponding correlation, while NA means that the correlation will be estimated. For example, with struct="CS" and rho=0, the variance-covariance matrix of the inner factor will be diagonal with $\tau^2$ along the diagonal. For struct="UN", the values specified under rho should be given in column-wise order (e.g., for an inner grouping variable/factor with four levels, the order would be $\rho_{12}$, $\rho_{13}$, $\rho_{23}$, $\rho_{14}$, $\rho_{24}$, $\rho_{34}$).

### Omnibus Test of Parameters

For models including moderators, an omnibus test of all the model coefficients is conducted that excludes the intercept (the first coefficient) if it is included in the model. If no intercept is included in the model, then the omnibus test includes all of the coefficients in the model including the first. Alternatively, one can manually specify the indices of the coefficients to test via the btt argument. For example, with btt=c(3,4), only the third and fourth coefficient from the model would be included in the test (if an intercept is included in the model, then it corresponds to the first coefficient in the model).

### Categorical Moderators

Categorical moderator variables can be included in the model via the mods argument in the same way that appropriately (dummy) coded categorical independent variables can be included in linear models. One can either do the dummy coding manually or use a model formula together with the factor function to let R handle the coding automatically.

### Tests and Confidence Intervals

By default, the test statistics of the individual coefficients in the model (and the corresponding confidence intervals) are based on the normal distribution, while the omnibus test is based on a chi-square distribution with $m$ degrees of freedom ($m$ being the number of coefficients tested). As an alternative, one can set tdist=TRUE, which slightly mimics the Knapp and Hartung (2003) method by using a t-distribution with $k - p$ degrees of freedom for tests of individual coefficients and confidence intervals and an F-distribution with $m$ and $k - p$ degrees of freedom ($p$ being the total number of model coefficients including the intercept if it is present) for the omnibus test statistic.

### Test for (Residual) Heterogeneity

A test for (residual) heterogeneity is automatically carried out by the function. Without moderators in the model, this test is the generalized/weighted least squares extension of Cochran's $Q$-test, which tests whether the variability in the observed effect sizes or outcomes is larger than one would expect based on sampling variability (and the given covariances among the sampling errors) alone. A significant test suggests that the true effects or outcomes are heterogeneous. When moderators are included in the model, this is the $Q_E$-test for residual heterogeneity, which tests whether the variability in the observed effect sizes or outcomes that is not accounted for by the moderators included in the model is larger than one would expect based on sampling variability (and the given covariances among the sampling errors) alone.

## Value

An object of class c("rma.mv","rma"). The object is a list containing the following components:

| | |
|---|---|
| b | estimated coefficients of the model. |
| se | standard errors of the coefficients. |
| zval | test statistics of the coefficients. |
| pval | p-values for the test statistics. |

| | |
|---|---|
| ci.lb | lower bound of the confidence intervals for the coefficients. |
| ci.ub | upper bound of the confidence intervals for the coefficients. |
| vb | variance-covariance matrix of the estimated coefficients. |
| sigma2 | estimated $\sigma^2$ value(s). |
| tau2 | estimated $\tau^2$ value(s). |
| rho | estimated $\rho$ value(s). |
| k | number of studies included in the model. |
| p | number of coefficients in the model (including the intercept). |
| m | number of coefficients included in the omnibus test of coefficients. |
| QE | test statistic for the test of (residual) heterogeneity. |
| QEp | p-value for the test of (residual) heterogeneity. |
| QM | test statistic for the omnibus test of coefficients. |
| QMp | p-value for the omnibus test of coefficients. |
| int.only | logical that indicates whether the model is an intercept-only model. |
| yi, V, X | the vector of outcomes, the corresponding variance-covariance matrix of the sampling errors, and the model matrix. |
| M | the estimated marginal variance-covariance matrix of the observed outcomes. |
| fit.stats | a list with the log-likelihood, deviance, AIC, BIC, and AICc values. |
| ... | some additional elements/values. |

The results of the fitted model are formated and printed with the `print.rma.mv` function. If fit statistics should also be given, use `summary.rma` (or use the `fitstats.rma` function to extract them).

Predicted/fitted values can be obtained with `predict.rma` and `fitted.rma`.

The `residuals.rma` and `rstandard.rma.mv` functions extract raw and standardized residuals.

For random/mixed-effects models, the `profile.rma.mv` function can be used to obtain a plot of the (restricted) log-likelihood as a function of a specific variance component or correlation parameter of the model.

Other extractor functions include `coef.rma`, `vcov.rma`, `logLik.rma`, `deviance.rma`, `AIC.rma`, `BIC.rma`, `hatvalues.rma.mv`, and `weights.rma.mv`.

### Note

Argument V also accepts a list of variance-covariance matrices for the observed effects or outcomes. From the list elements, the full (block diagonal) variance-covariance matrix is then automatically constructed.

Model fitting is done via numerical optimization over the model parameters. By default, `optim` is used with method `"Nelder-Mead"`. A different method can be chosen via the `control` argument (e.g., `control=list(optmethod="BFGS")`). One can also chose a different optimizer via the `optimizer` argument (so the default is `control=list(optimizer="optim")`). Besides `optim`, one can also choose `nlminb`, one of the optimizers from the minqa package (i.e., `uobyqa`, `newuoa`,

or `bobyqa`), or one of the (derivative-free) algorithms from the `nloptr` package. Note that the optimizer name should be given as a character string (i.e., in quotes). Additional control parameters for the optimizer can be specified via the `control` argument (e.g., `control=list(optimizer="nlminb", iter.max=500, rel` For `nloptr`, the default is to use the BOBYQA implementation from that package with a relative convergence criterion of `1e-8` on the function value (i.e., log likelihood), but this can be changed via the `algorithm` and `ftop_rel` arguments (e.g., `control=list(optimizer="nloptr", algorithm="NLOPT_LN_SBPLX", ft`

At the moment, the starting values are not chosen in a terribly clever way and may be far off. As a result, the optimizer may be slow to converge or may even get stuck in a local maximum. One can set the starting values manually for the various variance components and correlations in the model via the `control` argument by specifying the vectors `sigma2.init`, `tau2.init`, and/or `rho.init` as needed. Especially for complex models, it is a good idea to try out different starting values to make sure that the same estimates are obtained.

Information on the progress of the optimization algorithm can be obtained by setting `verbose=TRUE` or with `control=list(verbose=TRUE)`.

Whether particular variance components and/or correlations are actually identifiable needs to be carefully examined when fitting complex models. The function does some limited checking internally to fix variances and/or correlations at zero when it is clear that insufficient information is available to estimate a particular parameter (e.g., if a particular factor has only a single level, the corresponding variance component cannot be estimated). However, it is strongly advised in general to do post model fitting checks to make sure that the likelihood surface around the ML/REML estimates is not flat for some combination of the parameter estimates (which would imply that the estimates are essentially arbitrary). For example, one can plot the (restricted) log-likelihood as a function of each variance component and correlation in the model to make sure that each profile plot shows a clear peak at the corresponding ML/REML estimates. The `profile.rma.mv` function can be used for this purpose.

Finally, note that the model fitting is not done in a very efficient manner at the moment, which is partly a result of allowing for crossed random effects and correlations among the effect sizes or outcomes across the entire dataset (when using the R argument). As a result, the function works directly with the entire $k \times k$ (marginal) variance-covariance matrix of the effect sizes or outcomes (instead of working with smaller blocks in a block diagonal structure). As a result, model fitting can be slow for large $k$. However, when the variance-covariance structure is actually sparse, a lot of speed can be gained by setting `sparse=TRUE`. Also, when model fitting appears to be slow, setting `verbose=TRUE` is again useful to obtain information on how the model fitting is progressing.

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Berkey, C. S., Hoaglin, D. C., Antczak-Bouckoms, A., Mosteller, F., & Colditz, G. A. (1998). Meta-analysis of multiple outcomes by regression with random effects. *Statistics in Medicine*, **17**, 2537–2550.

Gleser, L. J., & Olkin, I. (2009). Stochastically dependent effect sizes. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 357–376). New York: Russell Sage Foundation.

van Houwelingen, H. C., Arends, L. R., & Stijnen, T. (2002). Advanced methods in meta-analysis: Multivariate approach and meta-regression. *Statistics in Medicine*, **21**, 589–624.

Ishak, K. J., Platt, R. W., Joseph, L., Hanley, J. A., & Caro, J. J. (2007). Meta-analysis of longitudinal studies. *Clinical Trials*, **4**, 525–539.

Kalaian, H. A., & Raudenbush, S. W. (1996). A multivariate mixed linear model for meta-analysis. *Psychological Methods*, **1**, 227-235.

Konstantopoulos, S. (2011). Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods*, **2**, 61–76.

Nakagawa, S., & Santos, E. S. A. (2012). Methodological issues and advances in biological meta-analysis. *Evolutionary Ecology*, **26**, 1253–1274.

Steiger, J. H. (1980). Tests for comparing elements of a correlation matrix. *Psychological Bulletin*, **87**, 245–251.

Salanti, G., Higgins, J. P. T., Ades, A. E., & Ioannidis, J. P. A. (2008). Evaluation of networks of randomized trials. *Statistical Methods in Medical Research*, **17**, 279–301.

Trikalinos, T. A., & Olkin, I. (2012). Meta-analysis of effect sizes reported at multiple time points: A multivariate approach. *Clinical Trials*, **9**, 610–620.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.mh, rma.peto, and rma.glmm for other model fitting functions.

dat.konstantopoulos2011, dat.hasselblad1998, dat.begg1989, dat.berkey1998, dat.fine1993, and dat.ishak2007 for further examples of the use of the rma.mv function.

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log odds ratios and corresponding sampling variances
dat <- escalc(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model using rma.uni()
rma(yi, vi, data=dat)

### random-effects model using rma.mv()
### note: sigma^2 in this model is the same as tau^2 from the previous model
rma.mv(yi, vi, random = ~ 1 | trial, data=dat)

### change data into long format
dat.long <- to.long(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### set levels of group variable ("exp" = experimental/vaccinated; "con" = control/non-vaccinated)
levels(dat.long$group) <- c("exp", "con")

### set "con" to reference level
dat.long$group <- relevel(dat.long$group, ref="con")
```

```
### calculate log odds and corresponding sampling variances
dat.long <- escalc(measure="PLO", xi=out1, mi=out2, data=dat.long)

### bivariate random-effects model using rma.mv()
res <- rma.mv(yi, vi, mods = ~ group, random = ~ group | study, struct="UN", data=dat.long)
res
```

rma.peto                          *Meta-Analysis via Peto's Method*

## Description

Function to fit fixed-effects models to $2 \times 2$ table data via Peto's method. See below and the documentation of the **metafor-package** for more details on these models.

## Usage

```
rma.peto(ai, bi, ci, di, n1i, n2i,
         data, slab, subset,
         add=1/2, to="only0", drop00=TRUE,
         level=95, digits=4, verbose=FALSE)
```

## Arguments

| | |
|---|---|
| ai | vector to specify the $2 \times 2$ table frequencies (upper left cell). See below and the documentation of the escalc function for more details. |
| bi | vector to specify the $2 \times 2$ table frequencies (upper right cell). See below and the documentation of the escalc function for more details. |
| ci | vector to specify the $2 \times 2$ table frequencies (lower left cell). See below and the documentation of the escalc function for more details. |
| di | vector to specify the $2 \times 2$ table frequencies (lower right cell). See below and the documentation of the escalc function for more details. |
| n1i | vector to specify the group sizes or row totals (first group). See below and the documentation of the escalc function for more details. |
| n2i | vector to specify the group sizes or row totals (second group). See below and the documentation of the escalc function for more details. |
| data | optional data frame containing the data supplied to the function. |
| slab | optional vector with labels for the $k$ studies. |
| subset | optional vector indicating the subset of tables that should be used for the analysis. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the tables to include. |

add             non-negative number indicating the amount to add to zero cells, counts, or frequencies when calculating the observed outcomes of the individual studies. Can also be a vector of two numbers, where the first number is used in the calculation of the observed outcomes and the second number is used when applying Peto's method. See below and the documentation of the [escalc](escalc) function for more details.

to              character string indicating when the values under add should be added (either "only0", "all", "if0all", or "none"). Can also be a character vector, where the first string again applies when calculating the observed outcomes and the second string when applying Peto's method. See below and the documentation of the [escalc](escalc) function for more details.

drop00          logical indicating whether studies with no cases (or only cases) in both groups should be dropped when calculating the observed outcomes (the outcomes for such studies are set to NA). Can also be a vector of two logicals, where the first applies to the calculation of the observed outcomes and the second when applying Peto's method. See below and the documentation of the [escalc](escalc) function for more details.

level           numerical value between 0 and 100 specifying the confidence interval level (default is 95).

digits          integer specifying the number of decimal places to which the printed results should be rounded (default is 4).

verbose         logical indicating whether output should be generated on the progress of the model fitting (default is FALSE).

### Details

#### Specifying the Data

The studies are assumed to provide data in terms of $2 \times 2$ tables of the form:

|         | outcome 1 | outcome 2 | total |
|---------|-----------|-----------|-------|
| group 1 | ai        | bi        | n1i   |
| group 2 | ci        | di        | n2i   |

where ai, bi, ci, and di denote the cell frequencies and n1i and n2i the row totals. For example, in a set of randomized clinical trials (RCTs) or cohort studies, group 1 and group 2 may refer to the treatment (exposed) and placebo/control (not exposed) group, with outcome 1 denoting some event of interest (e.g., death) and outcome 2 its complement. In a set of case-control studies, group 1 and group 2 may refer to the group of cases and the group of controls, with outcome 1 denoting, for example, exposure to some risk factor and outcome 2 non-exposure.

#### Peto's Method

An approach for aggregating $2 \times 2$ table data of this type was suggested by Peto (see Yusuf et al., 1985). The method provides a weighted estimate of the (log) odds ratio under a fixed-effects model. The method is particularly advantageous when the event of interest is rare, but it should only be used when the group sizes within the individual studies are not too dissimilar and effect sizes are generally small (Greenland & Salvan, 1990; Sweeting et al., 2004; Bradburn et al., 2007). Note that the printed results are given both in terms of the log and the raw units (for easier interpretation).

### Observed Outcomes of the Individual Studies

Peto's method itself does not require the calculation of the observed (log) odds ratios of the individual studies and directly makes use of the $2 \times 2$ table counts. Zero cells are not a problem (except in extreme cases, such as when one of the two outcomes never occurs in any of the tables). Therefore, it is unnecessary to add some constant to the cell counts when there are zero cells.

However, for plotting and various other functions, it is necessary to calculate the observed (log) odds ratios for the $k$ studies. Here, zero cells can be problematic, so adding a constant value to the cell counts ensures that all $k$ values can be calculated. The add and to arguments are used to specify what value should be added to the cell frequencies and under what circumstances when calculating the observed (log) odds ratios and when applying Peto's method. Similarly, the drop00 argument is used to specify how studies with no cases (or only cases) in both groups should be handled. The documentation of the [escalc](#) function explains how the add, to, and drop00 arguments work. If only a single value for these arguments is specified (as per default), then these values are used when calculating the observed (log) odds ratios and no adjustment to the cell counts is made when applying Peto's method. Alternatively, when specifying two values for these arguments, the first value applies when calculating the observed (log) odds ratios and the second value when applying Peto's method.

Note that drop00 is set to TRUE by default. Therefore, the observed (log) odds ratios for studies where ai=ci=0 or bi=di=0 are set to NA. When applying Peto's method, such studies are not explicitly dropped (unless the second value of drop00 argument is also set to TRUE), but this is practically not necessary, as they do not actually influence the results (assuming no adjustment to the cell/event counts are made when applying Peto's method).

### Value

An object of class c("rma.peto","rma"). The object is a list containing the following components:

| | |
|---|---|
| b | aggregated log odds ratio. |
| se | standard error of the aggregated value. |
| zval | test statistics of the aggregated value. |
| pval | p-value for the test statistic. |
| ci.lb | lower bound of the confidence interval. |
| ci.ub | upper bound of the confidence interval. |
| QE | test statistic for the test of heterogeneity. |
| QEp | p-value for the test of heterogeneity. |
| k | number of tables included in the analysis. |
| yi, vi | the vector of individual log odds ratios and corresponding sampling variances. |
| fit.stats | a list with the log-likelihood, deviance, AIC, BIC, and AICc values under the unrestricted and restricted likelihood. |
| ... | some additional elements/values. |

The results of the fitted model are formated and printed with the [print.rma.peto](#) function. If fit statistics should also be given, use [summary.rma](#) (or use the [fitstats.rma](#) function to extract them).

The `residuals.rma`, `rstandard.rma.peto`, and `rstudent.rma.peto` functions extract raw and standardized residuals. Leave-one-out diagnostics can be obtained with `leave1out.rma.peto`.

Forest, funnel, radial, L'abbé, and Baujat plots can be obtained with `forest.rma`, `funnel.rma`, `radial.rma`, `labbe.rma`, and `baujat.rma.peto`. The `qqnorm.rma.peto` function provides normal QQ plots of the standardized residuals. One can also just call `plot.rma.peto` on the fitted model object to obtain various plots at once.

A cumulative meta-analysis (i.e., adding one obervation at a time) can be obtained with `cumul.rma.peto`.

Other extractor functions include `coef.rma`, `vcov.rma`, `logLik.rma`, `deviance.rma`, `AIC.rma`, and `BIC.rma`.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Bradburn, M. J., Deeks, J. J., Berlin, J. A., & Localio, A. R. (2007). Much ado about nothing: A comparison of the performance of meta-analytical methods with rare events. *Statistics in Medicine*, **26**, 53–77.

Greenland, S., & Salvan, A. (1990). Bias in the one-step method for pooling study results. *Statistics in Medicine*, **9**, 247–252.

Sweeting, M. J., Sutton, A. J., & Lambert, P. C. (2004). What to add to nothing? Use and avoidance of continuity corrections in meta-analysis of sparse data. *Statistics in Medicine*, **23**, 1351–1375.

Yusuf, S., Peto, R., Lewis, J., Collins, R., & Sleight, P. (1985). Beta blockade during and after myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Disease*, **27**, 335–371.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

`rma.uni`, `rma.glmm`, `rma.mh`, and `rma.mv` for other model fitting functions.

`dat.collins1985a`, `dat.collins1985b`, and `dat.yusuf1985` for further examples of the use of the `rma.peto` function.

## Examples

```
### load data
data(dat.bcg)

### meta-analysis of the (log) odds ratios using Peto's method
rma.peto(ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
```

## rma.uni                          *Meta-Analysis via Linear (Mixed-Effects) Models*

### Description

Function to fit the meta-analytic fixed- and random/mixed-effects models with or without moderators via linear (mixed-effects) models. See the documentation of the **metafor-package** for more details on these models.

### Usage

```
rma.uni(yi, vi, sei, weights, ai, bi, ci, di, n1i, n2i, x1i, x2i,
        t1i, t2i, m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, sdi, ni, mods,
        measure="GEN", intercept=TRUE, data, slab, subset,
        add=1/2, to="only0", drop00=FALSE, vtype="LS",
        method="REML", weighted=TRUE, knha=FALSE,
        level=95, digits=4, btt, tau2, verbose=FALSE, control)
rma(yi, vi, sei, weights, ai, bi, ci, di, n1i, n2i, x1i, x2i,
        t1i, t2i, m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, sdi, ni, mods,
        measure="GEN", intercept=TRUE, data, slab, subset,
        add=1/2, to="only0", drop00=FALSE, vtype="LS",
        method="REML", weighted=TRUE, knha=FALSE,
        level=95, digits=4, btt, tau2, verbose=FALSE, control)
```

### Arguments

| | |
|---|---|
| yi | vector of length $k$ with the observed effect sizes or outcomes. See 'Details'. |
| vi | vector of length $k$ with the corresponding sampling variances. See 'Details'. |
| sei | vector of length $k$ with the corresponding standard errors (only relevant when not using vi). See 'Details'. |
| weights | optional argument to specify a vector of length $k$ with user-defined weights. See 'Details'. |
| ai | see below and the documentation of the escalc function for more details. |
| bi | see below and the documentation of the escalc function for more details. |
| ci | see below and the documentation of the escalc function for more details. |
| di | see below and the documentation of the escalc function for more details. |
| n1i | see below and the documentation of the escalc function for more details. |
| n2i | see below and the documentation of the escalc function for more details. |
| x1i | see below and the documentation of the escalc function for more details. |
| x2i | see below and the documentation of the escalc function for more details. |
| t1i | see below and the documentation of the escalc function for more details. |
| t2i | see below and the documentation of the escalc function for more details. |
| m1i | see below and the documentation of the escalc function for more details. |

| | |
|---|---|
| m2i | see below and the documentation of the [escalc](#) function for more details. |
| sd1i | see below and the documentation of the [escalc](#) function for more details. |
| sd2i | see below and the documentation of the [escalc](#) function for more details. |
| xi | see below and the documentation of the [escalc](#) function for more details. |
| mi | see below and the documentation of the [escalc](#) function for more details. |
| ri | see below and the documentation of the [escalc](#) function for more details. |
| ti | see below and the documentation of the [escalc](#) function for more details. |
| sdi | see below and the documentation of the [escalc](#) function for more details. |
| ni | see below and the documentation of the [escalc](#) function for more details. |
| mods | optional argument to include one or more moderators in the model. A single moderator can be given as a vector of length $k$ specifying the values of the moderator. Multiple moderators are specified by giving a matrix with $k$ rows and as many columns as there are moderator variables. Alternatively, a model [formula](#) can be used to specify the model. See 'Details'. |
| measure | character string indicating the type of data supplied to the function. When measure="GEN" (default), the observed effect sizes or outcomes and corresponding sampling variances (or standard errors) should be supplied to the function via the yi, vi, and sei arguments (only one of the two, vi or sei, needs to be specified). Alternatively, one can set measure to one of the effect size or outcome measures described under the documentation for the [escalc](#) function and specify the needed data via the appropriate arguments. |
| intercept | logical indicating whether an intercept term should be added to the model (default is TRUE). |
| data | optional data frame containing the data supplied to the function. |
| slab | optional vector with labels for the $k$ studies. |
| subset | optional vector indicating the subset of studies that should be used for the analysis. This can be a logical vector of length $k$ or a numeric vector indicating the indices of the observations to include. |
| add | see the documentation of the [escalc](#) function. |
| to | see the documentation of the [escalc](#) function. |
| drop00 | see the documentation of the [escalc](#) function. |
| vtype | see the documentation of the [escalc](#) function. |
| method | character string specifying whether a fixed- or a random/mixed-effects model should be fitted. A fixed-effects model (with or without moderators) is fitted when using method="FE". Random/mixed-effects models are fitted by setting method equal to one of the following: "DL", "HE", "SJ", "ML", "REML", "EB", "HS", or "GENQ". Default is "REML". See 'Details'. |
| weighted | logical indicating whether weighted (default) or unweighted estimation should be used to fit the model. |
| knha | logical specifying whether the method by Knapp and Hartung (2003) should be used for adjusting test statistics and confidence intervals (default is FALSE). See 'Details'. |

| level | numerical value between 0 and 100 specifying the confidence interval level (default is 95). |
| --- | --- |
| digits | integer specifying the number of decimal places to which the printed results should be rounded (default is 4). |
| btt | optional vector of indices specifying which coefficients to include in the omnibus test of moderators. See 'Details'. |
| tau2 | optional numerical value to specify the amount of (residual) heterogeneity in a random- or mixed-effects model (instead of estimating it). Useful for sensitivity analyses (e.g., for plotting results as a function of $\tau^2$). When unspecified, the value of $\tau^2$ is estimated from the data. |
| verbose | logical indicating whether output should be generated on the progress of the model fitting (default is FALSE). Can also be an integer. Values > 1 generate more verbose output. See 'Note'. |
| control | optional list of control values for the iterative estimation algorithms. If unspecified, default values are defined inside the function. See 'Note'. |

### Details

#### Specifying the Data

The function can be used in conjunction with any of the usual effect size or outcome measures used in meta-analyses (e.g., log odds ratios, log relative risks, risk differences, mean differences, standardized mean differences, raw correlation coefficients, correlation coefficients transformed with Fisher's r-to-z transformation, and so on). Simply specify the observed outcomes via the yi argument and the corresponding sampling variances via the vi argument. Instead of specifying vi, one can specify the standard errors (the square root of the sampling variances) via the sei argument. The [escalc](#) function can be used to compute a wide variety of effect size and outcome measures (and the corresponding sampling variances) based on summary statistics.

Alternatively, the function can automatically calculate the values of a chosen effect size or outcome measure (and the corresponding sampling variances) when supplied with the necessary data. The [escalc](#) function describes which effect size or outcome measures are currently implemented and what data/arguments should then be specified/used. The measure argument should then be set to the desired effect size or outcome measure.

#### Specifying the Model

The function can be used to fit fixed- and random/mixed-effects models, as well as meta-regression models including moderators (the difference between the various models is described in detail in the introductory **metafor-package** help file).

Assuming the observed outcomes and corresponding sampling variances are supplied via yi and vi, a *fixed-effects model* can be fitted with rma(yi, vi, method="FE"). Weighted estimation (with inverse-variance weights) is used by default. User-defined weights can be supplied via the weights argument. Unweighted estimation can be used by setting weighted=FALSE (same as setting the weights equal to a constant).

A *random-effects model* can be fitted with the same code but setting method to one of the various estimators for the amount of heterogeneity:

- method="DL" = DerSimonian-Laird estimator

- method="HE" = Hedges estimator

- method="HS" = Hunter-Schmidt estimator

- method="SJ" = Sidik-Jonkman estimator

- method="ML" = maximum-likelihood estimator

- method="REML" = restricted maximum-likelihood estimator

- method="EB" = empirical Bayes (= Paule-Mandel) estimator

- method="GENQ" = generalized Q-statistic estimator

For a description of the various estimators, see DerSimonian and Kacker (2007), Raudenbush (2009), Viechtbauer (2005), and Viechtbauer et al. (in press). Note that the Hedges estimator is also called the 'variance component estimator' or 'Cochran estimator', the Sidik-Jonkman estimator is also called the 'model error variance estimator', and the empirical Bayes estimator is actually identical to the Paule-Mandel estimator (Paule & Mandel, 1982). Finally, the generalized Q-statistic estimator is a general method-of-moments estimator requiring the specification of weights (the HE and DL estimators are just special cases with equal and inverse variance weights, respectively).

One or more moderators can be included in these models via the mods argument. A single moderator can be given as a (row or column) vector of length $k$ specifying the values of the moderator. Multiple moderators are specified by giving an appropriate model matrix (i.e., $\mathbf{X}$) with $k$ rows and as many columns as there are moderator variables (e.g., mods = cbind(mod1, mod2, mod3), where mod1, mod2, and mod3 correspond to the names of the variables for three moderator variables). The intercept is added to the model matrix by default unless intercept=FALSE.

Alternatively, one can use the standard [formula](formula) syntax to specify the model. In this case, the mods argument should be set equal to a one-sided formula of the form mods = ~ model (e.g., mods = ~ mod1 + mod2 + mod3). Interactions, polynomial terms, and factors can be easily added to the model in this manner. When specifying a model formula via the mods argument, the intercept argument is ignored. Instead, the inclusion/exclusion of the intercept term is controlled by the specified formula (e.g., mods = ~ mod1 + mod2 + mod3 - 1 would lead to the removal of the intercept term).

A *fixed-effects with moderators model* is then fitted by setting method="FE", while a *mixed-effects model* is fitted by specifying one of the estimators for the amount of (residual) heterogeneity given earlier.

When the observed outcomes and corresponding sampling variances are supplied via the yi and vi (or sei) arguments, one can also directly specify moderators via the yi argument (e.g., rma(yi ~ mod1 + mod2 + mod3, vi)). In that case, the mods argument is ignored and the inclusion/exclusion of the intercept term again is controlled by the specified formula.

**Omnibus Test of Parameters**

For models including moderators, an omnibus test of all the model coefficients is conducted that excludes the intercept (the first coefficient) if it is included in the model. If no intercept is included in the model, then the omnibus test includes all of the coefficients in the model including the first. Alternatively, one can manually specify the indices of the coefficients to test via the btt argument. For example, with btt=c(3,4), only the third and fourth coefficient from the model would be included in the test (if an intercept is included in the model, then it corresponds to the first coefficient in the model).

**Categorical Moderators**

Categorical moderator variables can be included in the model via the mods argument in the same way that appropriately (dummy) coded categorical independent variables can be included in linear models. One can either do the dummy coding manually or use a model formula together with the [factor](factor) function to let R handle the coding automatically. An example to illustrate these different approaches is provided below.

### Knapp & Hartung Adjustment

By default, the test statistics of the individual coefficients in the model (and the corresponding confidence intervals) are based on the normal distribution, while the omnibus test is based on a chi-square distribution with $m$ degrees of freedom ($m$ being the number of coefficients tested). The Knapp and Hartung (2003) method (knha=TRUE) is an adjustment to the standard errors of the estimated coefficients, which helps to account for the uncertainty in the estimate of the amount of (residual) heterogeneity and leads to different reference distributions. Tests of individual coefficients and confidence intervals are then based on the t-distribution with $k - p$ degrees of freedom, while the omnibus test statistic then uses an F-distribution with $m$ and $k - p$ degrees of freedom ($p$ being the total number of model coefficients including the intercept if it is present). The Knapp and Hartung (2003) adjustment is only meant to be used in the context of random- or mixed-effects models.

### Test for (Residual) Heterogeneity

A test for (residual) heterogeneity is automatically carried out by the function. Without moderators in the model, this is simply Cochran's $Q$-test (Cochran, 1954), which tests whether the variability in the observed effect sizes or outcomes is larger than would be expected based on sampling variability alone. A significant test suggests that the true effects or outcomes are heterogeneous. When moderators are included in the model, this is the $Q_E$-test for residual heterogeneity, which tests whether the variability in the observed effect sizes or outcomes not accounted for by the moderators included in the model is larger than would be expected based on sampling variability alone.

### Value

An object of class c("rma.uni","rma"). The object is a list containing the following components:

| | |
|---|---|
| b | estimated coefficients of the model. |
| se | standard errors of the coefficients. |
| zval | test statistics of the coefficients. |
| pval | p-values for the test statistics. |
| ci.lb | lower bound of the confidence intervals for the coefficients. |
| ci.ub | upper bound of the confidence intervals for the coefficients. |
| vb | variance-covariance matrix of the estimated coefficients. |
| tau2 | estimated amount of (residual) heterogeneity. Always 0 when method="FE". |
| se.tau2 | estimated standard error of the estimated amount of (residual) heterogeneity. |
| k | number of outcomes included in the model fitting. |
| p | number of coefficients in the model (including the intercept). |
| m | number of coefficients included in the omnibus test of coefficients. |
| QE | test statistic for the test of (residual) heterogeneity. |
| QEp | p-value for the test of (residual) heterogeneity. |

| QM | test statistic for the omnibus test of coefficients. |
|---|---|
| QMp | p-value for the omnibus test of coefficients. |
| I2 | value of $I^2$. See `print.rma.uni` for more details. |
| H2 | value of $H^2$. See `print.rma.uni` for more details. |
| R2 | value of $R^2$. See `print.rma.uni` for more details. |
| int.only | logical that indicates whether the model is an intercept-only model. |
| yi, vi, X | the vector of outcomes, the corresponding sampling variances, and the model matrix. |
| fit.stats | a list with the log-likelihood, deviance, AIC, BIC, and AICc values under the unrestricted and restricted likelihood. |
| ... | some additional elements/values. |

The results of the fitted model are formated and printed with the `print.rma.uni` function. If fit statistics should also be given, use `summary.rma` (or use the `fitstats.rma` function to extract them). Full versus reduced model comparisons in terms of fit statistics and likelihoods can be obtained with `anova.rma`. Wald-type tests for sets of model coefficients can be obtained with the same function. Permutation tests for the model coefficient(s) can be obtained with `permutest.rma.uni`.

Predicted/fitted values can be obtained with `predict.rma` and `fitted.rma`. For best linear unbiased predictions, see `blup.rma.uni`.

The `residuals.rma`, `rstandard.rma.uni`, and `rstudent.rma.uni` functions extract raw and standardized residuals. Additional case diagnostics (e.g., to determine influential studies) can be obtained with the `influence.rma.uni` function. For models without moderators, leave-one-out diagnostics can also be obtained with `leave1out.rma.uni`.

A confidence interval for the amount of (residual) heterogeneity in the random/mixed-effects model can be obtained with `confint.rma.uni`.

Forest, funnel, radial, L'abbé, and Baujat plots can be obtained with `forest.rma`, `funnel.rma`, `radial.rma`, `labbe.rma`, and `baujat.rma.uni` (radial and L'abbé plots only for models without moderators). The `qqnorm.rma.uni` function provides normal QQ plots of the standardized residuals. One can also just call `plot.rma.uni` on the fitted model object to obtain various plots at once. For random/mixed-effects models, the `profile.rma.uni` function can be used to obtain a plot of the (restricted) log-likelihood as a function of $\tau^2$.

Tests for funnel plot asymmetry (which may be indicative of publication bias) can be obtained with `ranktest.rma` and `regtest.rma`. For models without moderators, the `trimfill.rma.uni` method can be used to carry out a trim and fill analysis and `hc.rma.uni` provides a random-effects model analysis that is more robust to publication bias (based on the method by Henmi & Copas, 2010).

For models without moderators, a cumulative meta-analysis (i.e., adding one obervation at a time) can be obtained with `cumul.rma.uni`.

Other extractor functions include `coef.rma`, `vcov.rma`, `logLik.rma`, `deviance.rma`, `AIC.rma`, `BIC.rma`, `hatvalues.rma.uni`, and `weights.rma.uni`.

## Note

While the HS, HE, DL, SJ, and GENQ estimators of $\tau^2$ are based on closed-form solutions, the ML, REML, and EB estimators must be obtained numerically. For this, the `rma()` function makes use of

the Fisher scoring algorithm, which is robust to poor starting values and usually converges quickly (Harville, 1977; Jennrich & Sampson, 1976). By default, the starting value is set equal to the value of the Hedges estimator and the algorithm terminates when the change in the estimated value of $\tau^2$ is smaller than $10^{-5}$ from one iteration to the next. The maximum number of iterations is 100 by default (which should be sufficient in most cases). Information on the progress of the algorithm can be obtained by setting `verbose=TRUE` or with `control=list(verbose=TRUE)`.

A different starting value, threshold, and maximum number of iterations can be specified via the `control` argument by setting `control=list(tau2.init=value, threshold=value, maxiter=value)`. The step length of the Fisher scoring algorithm can also be manually adjusted by a desired factor with `control=list(stepadj=value)` (values below 1 will reduce the step length). If using `verbose=TRUE` shows the estimate jumping around erratically (or cycling through a few values), decreasing the step length can often help with convergence (e.g., `control=list(stepadj=0.5)`). At the same time, the maximum number of iterations may need to be increased.

All of the heterogeneity estimators except SJ can in principle yield negative estimates for the amount of (residual) heterogeneity. However, negative estimates of $\tau^2$ are outside of the parameter space. For the HS, HE, and DL estimators, negative estimates are therefore truncated to zero. For ML, REML, and EB estimation, the Fisher scoring algorithm makes use of step halving to guarantee a non-negative estimate. For those brave enough to step into risky territory, there is the option to set the lower bound of $\tau^2$ equal to some other value besides zero with `control=list(tau2.min=value)`.

The Hunter-Schmidt estimator for the amount of heterogeneity is defined in Hunter and Schmidt (1990) only in the context of the random-effects model when analyzing correlation coefficients. A general version of this estimator for the random-effects model not specific to any particular outcome measure is described in Viechtbauer (2005). The same idea can be easily extended to the mixed-effects model (see Viechtbauer et al., in press) and is implemented here.

The Sidik-Jonkman estimator starts with a crude estimate of $\tau^2$, which is then updated as described in Sidik and Jonkman (2005b, 2007). If, instead of the crude estimate, one wants to use a better a priori estimate, one can do so by passing this value via `control=list(tau2.init=value)`.

Outcomes with non-positive sampling variances are problematic. If a sampling variance is equal to zero, then its weight will be $1/0$ for fixed-effects models when using weighted estimation. Switching to unweighted estimation is a possible solution then. For random/mixed-effects model, some estimators of $\tau^2$ are undefined when there is at least one sampling variance equal to zero. Other estimators may work, but it may still be necessary to switch to unweighted model fitting, especially when the estimate of $\tau^2$ turns out to be zero.

When including moderators in the model, it is possible that the model matrix is not of full rank (i.e., there is a linear relationship between the moderator variables included in the model). In that case, the model cannot be fitted and an error will be issued. For example, two moderators that correlate perfectly would cause this problem. Deleting (redundant) moderator variables from the model as needed should solve this problem.

Finally, some general words of caution about the assumptions underlying the models are warranted:

- The sampling variances (i.e., the `vi` values) are treated as if they are known constants. This (usually) implies that the distributions of the test statistics and corresponding confidence intervals are only exact and have nominal coverage when the within-study sample sizes are large (i.e., when the error in the sampling variance estimates is small). Certain outcome measures (e.g., the arcsine square-root transformed risk difference and Fisher's r-to-z transformed cor-

relation coefficient) are based on variance stabilizing transformations that also help to make the assumption of known sampling variances much more reasonable.

- When fitting a mixed/random-effects model, $\tau^2$ is estimated and then treated as a known constant thereafter. This ignores the uncertainty in the estimate of $\tau^2$. As a consequence, the standard errors of the parameter estimates tend to be too small, yielding test statistics that are too large and confidence intervals that are not wide enough. The Knapp and Hartung (2003) adjustment can be used to counter this problem, yielding test statistics and confidence intervals whose properties are closer to nominal.

- Most effect size measures are not exactly normally distributed as assumed under the various models. However, the normal approximation usually becomes more accurate for most effect size or outcome measures as the within-study sample sizes increase. Therefore, sufficiently large within-study sample sizes are (usually) needed to be certain that the tests and confidence intervals have nominal levels/coverage. Again, certain outcome measures (e.g., Fisher's r-to-z transformed correlation coefficient) may be preferable from this perspective as well.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Cochran, W. G. (1954). The combination of estimates from different experiments. *Biometrics*, **10**, 101–129.

DerSimonian, R., & Laird, N. (1986). Meta-analysis in clinical trials. *Controlled Clinical Trials*, **7**, 177–188.

DerSimonian, R., & Kacker, R. (2007). Random-effects model for meta-analysis of clinical trials: An update. *Contemporary Clinical Trials*, **28**, 105–114.

Harville, D. A. (1977). Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association*, **72**, 320–338.

Hedges, L. V. (1983). A random effects model for effect sizes. *Psychological Bulletin*, **93**, 388–395.

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. San Diego, CA: Academic Press.

Henmi, M., & Copas, J. B. (2010). Confidence intervals for random effects meta-analysis and robustness to publication bias. *Statistics in Medicine*, **29**, 2969–2983.

Hunter, J. E., & Schmidt, F. L. (2004). *Methods of meta-analysis: Correcting error and bias in research findings* (2nd ed.). Thousand Oaks, CA: Sage.

Jennrich, R. I., & Sampson, P. F. (1976). Newton-Raphson and related algorithms for maximum likelihood variance component estimation. *Technometrics*, **18**, 11–17.

Knapp, G., & Hartung, J. (2003). Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine*, **22**, 2693–2710.

Morris, C. N. (1983). Parametric empirical Bayes inference: Theory and applications (with discussion). *Journal of the American Statistical Association*, **78**, 47–65.

Paule, R. C., & Mandel, J. (1982). Consensus values and weighting factors. *Journal of Research of the National Bureau of Standards*, **87**, 377–385.

Raudenbush, S. W. (2009). Analyzing effect sizes: Random effects models. In H. Cooper, L. V. Hedges, & J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp. 295–315). New York: Russell Sage Foundation.

Sidik, K., & Jonkman, J. N. (2005a). A note on variance estimation in random effects meta-regression. *Journal of Biopharmaceutical Statistics*, **15**, 823–838.

Sidik, K., & Jonkman, J. N. (2005b). Simple heterogeneity variance estimation for meta-analysis. *Journal of the Royal Statistical Society, Series C*, **54**, 367–384.

Sidik, K., & Jonkman, J. N. (2007). A comparison of heterogeneity variance estimators in combining results of studies. *Statistics in Medicine*, **26**, 1964–1981.

Viechtbauer, W. (2005). Bias and efficiency of meta-analytic variance estimators in the random-effects model. *Journal of Educational and Behavioral Statistics*, **30**, 261–293.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

Viechtbauer, W., López-López, J. A., Sánchez-Meca, J., & Marín-Martínez, F. (in press). A comparison of procedures to test for moderators in mixed-effects meta-regression models. *Psychological Methods*.

## See Also

rma.mh, rma.peto, rma.glmm, and rma.mv for other model fitting functions.

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model (method="REML" is default, so technically not needed)
rma(yi, vi, data=dat, method="REML")
rma(yi, sei=sqrt(vi), data=dat, method="REML")

### supplying the 2x2 table cell frequencies directly to the rma() function
rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat, method="REML")

### mixed-effects model with two moderators (absolute latitude and publication year)
rma(yi, vi, mods=cbind(ablat, year), data=dat, method="REML")

### using a model formula to specify the same model
rma(yi, vi, mods = ~ ablat + year, data=dat, method="REML")

### using a model formula as part of the yi argument
rma(yi ~ ablat + year, vi, data=dat, method="REML")

### manual dummy coding of the allocation factor
alloc.random    <- ifelse(dat$alloc == "random",    1, 0)
```

```
alloc.alternate  <- ifelse(dat$alloc == "alternate",  1, 0)
alloc.systematic <- ifelse(dat$alloc == "systematic", 1, 0)

### test the allocation factor (in the presence of the other moderators)
### note: "alternate" is the reference level of the allocation factor
### note: the intercept is the first coefficient, so btt=c(2,3)
rma(yi, vi, mods = ~ alloc.random + alloc.systematic + year + ablat,
    data=dat, method="REML", btt=c(2,3))

### using a model formula to specify the same model
rma(yi, vi, mods = ~ factor(alloc) + year + ablat, data=dat, method="REML", btt=c(2,3))

### test all pairwise differences with Holm's method (using the 'multcomp' package if installed)
res <- rma(yi, vi, mods = ~ factor(alloc) - 1, data=dat, method="REML")
res
if (require(multcomp))
   summary(glht(res, linfct=contrMat(c("alternate"=1,"random"=1,"systematic"=1),
           type="Tukey")), test=adjusted("holm"))

### subgrouping versus using a single model with a factor (subgrouping provides
### an estimate of tau^2 within each subgroup, but the number of studies in each
### subgroup is quite small; the model with the allocation factor provides a
### single estimate of tau^2 based on a larger number of studies, but assumes
### that tau^2 is the same within each subgroup)
res.a <- rma(yi, vi, data=dat, subset=(alloc=="alternate"))
res.r <- rma(yi, vi, data=dat, subset=(alloc=="random"))
res.s <- rma(yi, vi, data=dat, subset=(alloc=="systematic"))
res.a
res.r
res.s
res <- rma(yi, vi, mods = ~ factor(alloc) - 1, data=dat)
res

### demonstrating that Q_E + Q_M = Q_Total for fixed-effects models
### note: this does not work for random/mixed-effects models, since Q_E and
### Q_Total are calculated under the assumption that tau^2 = 0, while the
### calculation of Q_M incorporates the estimate of tau^2
res <- rma(yi, vi, data=dat, method="FE")
res ### this gives Q_Total
res <- rma(yi, vi, mods = ~ ablat + year, data=dat, method="FE")
res ### this gives Q_E and Q_M
res$QE + res$QM

### decomposition of Q_E into subgroup Q-values
res <- rma(yi, vi, mods = ~ factor(alloc), data=dat)
res

res.a <- rma(yi, vi, data=dat, subset=(alloc=="alternate"))
res.r <- rma(yi, vi, data=dat, subset=(alloc=="random"))
res.s <- rma(yi, vi, data=dat, subset=(alloc=="systematic"))

res.a$QE ### Q-value within subgroup "alternate"
res.r$QE ### Q-value within subgroup "random"
```

```
res.s$QE ### Q-value within subgroup "systematic"

res$QE
res.a$QE + res.r$QE + res.s$QE
```

---

to.long                    *Convert Data from Vector to Long Format*

---

### Description

The function can be used to convert summary data in vector format to the corresponding long
format.

### Usage

```
to.long(measure, ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i,
        m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, sdi, ni, data, slab, subset,
        add=1/2, to="none", drop00=FALSE, vlong=FALSE, append=TRUE, var.names)
```

### Arguments

| | |
|---|---|
| measure | a character string indicating the effect size or outcome measure corresponding to the summary data supplied. See below and the documentation of the [escalc](#) function for more details. |
| ai | vector to specify the $2 \times 2$ table frequencies (upper left cell). |
| bi | vector to specify the $2 \times 2$ table frequencies (upper right cell). |
| ci | vector to specify the $2 \times 2$ table frequencies (lower left cell). |
| di | vector to specify the $2 \times 2$ table frequencies (lower right cell). |
| n1i | vector to specify the group sizes or row totals (first group/row). |
| n2i | vector to specify the group sizes or row totals (second group/row). |
| x1i | vector to specify the number of events (first group). |
| x2i | vector to specify the number of events (second group). |
| t1i | vector to specify the total person-times (first group). |
| t2i | vector to specify the total person-times (second group). |
| m1i | vector to specify the means (first group or time point). |
| m2i | vector to specify the means (second group or time point). |
| sd1i | vector to specify the standard deviations (first group or time point). |
| sd2i | vector to specify the standard deviations (second group or time point). |
| xi | vector to specify the frequencies of the event of interest. |
| mi | vector to specify the frequencies of the complement of the event of interest or the group means. |
| ri | vector to specify the raw correlation coefficients. |

| | |
|---|---|
| `ti` | vector to specify the total person-times. |
| `sdi` | vector to specify the standard deviations. |
| `ni` | vector to specify the sample/group sizes. |
| `data` | optional data frame containing the variables given to the arguments above. |
| `slab` | optional vector with labels for the studies. |
| `subset` | optional vector indicating the subset of studies that should be used. This can be a logical vector or a numeric vector indicating the indices of the studies to include. |
| `add` | see the documentation of the [escalc](#) function. |
| `to` | see the documentation of the [escalc](#) function. |
| `drop00` | see the documentation of the [escalc](#) function. |
| `vlong` | optional logical whether a very long format should be used (only relevant for $2 \times 2$ or $1 \times 2$ table data). |
| `append` | logical indicating whether the data frame specified via the `data` argument (if one has been specified) should be returned together with the long format data (default is TRUE). |
| `var.names` | optional vector with variable names (length depends on the data type). If unspecified, the function sets appropriate variable names by default. |

### Details

The [escalc](#) function describes a wide variety of effect size and outcome measures that can be computed for a meta-analysis. The summary data used to compute those measures are typically contained in vectors, each element corresponding to a study. The `to.long` function takes this information and constructs a long format dataset from these data.

For example, in various fields (such as the health and medical sciences), the response variable measured is often dichotomous (binary), so that the data from a study comparing two different groups can be expressed in terms of a $2 \times 2$ table, such as:

| | outcome 1 | outcome 2 | total |
|---|---|---|---|
| group 1 | ai | bi | n1i |
| group 2 | ci | di | n2i |

where `ai`, `bi`, `ci`, and `di` denote the cell frequencies (i.e., the number of people falling into a particular category) and `n1i` and `n2i` the row totals (i.e., the group sizes).

The cell frequencies in $k$ such $2 \times 2$ tables can be specified via the `ai`, `bi`, `ci`, and `di` arguments (or alternatively, via the `ai`, `ci`, `n1i`, and `n2i` arguments). The function then creates the corresponding long format dataset. The `measure` argument should then be set equal to one of the outcome measures that can be computed based on this type of data, such as ″RR″, ″OR″, ″RD″ (it is not relevant which specific measure is chosen, as long as it corresponds to the specified summary data). See the documentation of the [escalc](#) function for more details on the types of data formats available.

The long format for data of this type consists of two rows per study, a factor indicating the study (default name `study`), a dummy variable indicating the group (default name `group`, coded as 1 and 2), and two variables indicating the number of individuals experiencing outcome 1 or outcome 2

(default names out1 and out2). Alternatively, if vlong=TRUE, then the long format consists of four rows per study, a factor indicating the study (default name study), a dummy variable indicating the group (default name group, coded as 1 and 2), a dummy variable indicating the outcome (default name outcome, coded as 1 and 2), and a variable indicating the frequency of the respective outcome (default name freq).

The default variable names can be changed via the var.names argument (must be of the appropriate length, depending on the data type).

The examples below illustrate the use of this function.

**Value**

A data frame with either $k$, $2k$, or $4k$ rows and an appropriate number of columns (depending on the data type) with the data in long format. If append=TRUE and a data frame was specified via the data argument, then the data in long format are appended to the original data frame (with rows repeated an appropriate number of times).

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

**References**

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

escalc, to.table

**Examples**

```
### load BCG vaccine data
data(dat.bcg)

### convert data to long format
dat <- to.long(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
dat

### extra long format
dat <- to.long(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg, vlong=TRUE)
dat

### load data from Hart et al. (1999)
data(dat.hart1999)

### convert data to long format
dat <- to.long(measure="IRR", x1i=x1i, x2i=x2i, t1i=t1i, t2i=t2i,
               data=dat.hart1999, var.names=c("id", "group", "events", "ptime"))
```

```
dat

### load data from Normand (1999)
data(dat.normand1999)

### convert data to long format
dat <- to.long(measure="MD", m1i=m1i, sd1i=sd1i, n1i=n1i,
               m2i=m2i, sd2i=sd2i, n2i=n2i, data=dat.normand1999,
               var.names=c("id", "group", "mean", "sd", "n"))
dat
```

---

| to.table | *Convert Data from Vector to Table Format* |
|---|---|

---

## Description

The function can be used to convert summary data in vector format to the corresponding table format.

## Usage

```
to.table(measure, ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i,
         m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, sdi, ni, data, slab, subset,
         add=1/2, to="none", drop00=FALSE, rows, cols)
```

## Arguments

| | |
|---|---|
| measure | a character string indicating the effect size or outcome measure corresponding to the summary data supplied. See below and the documentation of the [escalc](#) function for more details. |
| ai | vector to specify the $2 \times 2$ table frequencies (upper left cell). |
| bi | vector to specify the $2 \times 2$ table frequencies (upper right cell). |
| ci | vector to specify the $2 \times 2$ table frequencies (lower left cell). |
| di | vector to specify the $2 \times 2$ table frequencies (lower right cell). |
| n1i | vector to specify the group sizes or row totals (first group/row). |
| n2i | vector to specify the group sizes or row totals (second group/row). |
| x1i | vector to specify the number of events (first group). |
| x2i | vector to specify the number of events (second group). |
| t1i | vector to specify the total person-times (first group). |
| t2i | vector to specify the total person-times (second group). |
| m1i | vector to specify the means (first group or time point). |
| m2i | vector to specify the means (second group or time point). |
| sd1i | vector to specify the standard deviations (first group or time point). |
| sd2i | vector to specify the standard deviations (second group or time point). |

| | |
|---|---|
| xi | vector to specify the frequencies of the event of interest. |
| mi | vector to specify the frequencies of the complement of the event of interest or the group means. |
| ri | vector to specify the raw correlation coefficients. |
| ti | vector to specify the total person-times. |
| sdi | vector to specify the standard deviations. |
| ni | vector to specify the sample/group sizes. |
| data | optional data frame containing the variables given to the arguments above. |
| slab | optional vector with labels for the studies. |
| subset | optional vector indicating the subset of studies that should be used. This can be a logical vector or a numeric vector indicating the indices of the studies to include. |
| add | see the documentation of the [escalc](#) function. |
| to | see the documentation of the [escalc](#) function. |
| drop00 | see the documentation of the [escalc](#) function. |
| rows | optional vector with row/group names. |
| cols | optional vector with column/outcome names. |

### Details

The [escalc](#) function describes a wide variety of effect size and outcome measures that can be computed for a meta-analysis. The summary data used to compute those measures are typically contained in vectors, each element corresponding to a study. The to.table function takes this information and constructs an array of $k$ tables from these data.

For example, in various fields (such as the health and medical sciences), the response variable measured is often dichotomous (binary), so that the data from a study comparing two different groups can be expressed in terms of a $2 \times 2$ table, such as:

| | outcome 1 | outcome 2 | total |
|---|---|---|---|
| group 1 | ai | bi | n1i |
| group 2 | ci | di | n2i |

where ai, bi, ci, and di denote the cell frequencies (i.e., the number of people falling into a particular category) and n1i and n2i the row totals (i.e., the group sizes).

The cell frequencies in $k$ such $2 \times 2$ tables can be specified via the ai, bi, ci, and di arguments (or alternatively, via the ai, ci, n1i, and n2i arguments). The function then creates the corresponding $2 \times 2 \times k$ array of tables. The measure argument should then be set equal to one of the outcome measures that can be computed based on this type of data, such as ″RR″, ″OR″, ″RD″ (it is not relevant which specific measure is chosen, as long as it corresponds to the specified summary data). See the documentation of the [escalc](#) function for more details on the types of data formats available.

The examples below illustrate the use of this function.

## Value

An array with k elements each consisting of either 1 or 2 rows and an appropriate number of columns.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

escalc, to.long

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### create tables
dat <- to.table(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
                data=dat.bcg, slab=paste(author, year, sep=", "),
                rows=c("Vaccinated", "Not Vaccinated"), cols=c("TB+", "TB-"))
dat

### load data from Hart et al. (1999)
data(dat.hart1999)

### create tables
dat <- to.table(measure="IRR", x1i=x1i, x2i=x2i, t1i=t1i, t2i=t2i,
                data=dat.hart1999, slab=paste(study, year, sep=", "),
                rows=c("Warfarin Group", "Placebo/Control Group"))
dat

### load data from Normand (1999)
data(dat.normand1999)

### create tables
dat <- to.table(measure="MD", m1i=m1i, sd1i=sd1i, n1i=n1i,
                m2i=m2i, sd2i=sd2i, n2i=n2i, data=dat.normand1999,
                slab=source, rows=c("Specialized Care", "Routine Care"))
dat
```

---

transf                              *Transformation Function*

---

### Description

A set of transformation functions useful for meta-analyses.

### Usage

```
transf.rtoz(xi, ...)
transf.ztor(xi, ...)
transf.logit(xi, ...)
transf.ilogit(xi, ...)
transf.arcsin(xi, ...)
transf.iarcsin(xi, ...)
transf.pft(xi, ni, ...)
transf.ipft(xi, ni, ...)
transf.ipft.hm(xi, targs, ...)
transf.isqrt(xi, ...)
transf.irft(xi, ti, ...)
transf.iirft(xi, ti, ...)
transf.ahw(xi, ...)
transf.iahw(xi, ...)
transf.abt(xi, ...)
transf.iabt(xi, ...)
transf.ztor.int(xi, targs, ...)
transf.exp.int(xi, targs, ...)
transf.ilogit.int(xi, targs, ...)
```

### Arguments

| | |
|---|---|
| xi | vector of values to be transformed. |
| ni | vector of sample sizes. |
| ti | vector of person-times at risk. |
| targs | list with additional arguments for the transformation function. See 'Details'. |
| ... | other arguments. |

### Details

The following transformation functions are currently implemented:

- transf.rtoz: Fisher's r-to-z transformation for correlations.
- transf.ztor: inverse of the Fisher's r-to-z transformation.
- transf.logit: logit (log odds) transformation for proportions.
- transf.ilogit: inverse of the logit transformation.

- transf.arcsin: arcsine square-root transformation for proportions.

- transf.iarcsin: inverse of the arcsine transformation.

- transf.pft: Freeman-Tukey (double arcsine) transformation for proportions. See Freeman & Tukey (1950). The xi argument is used to specify the proportions and the ni argument the corresponding sample sizes.

- transf.ipft: inverse of the Freeman-Tukey (double arcsine) transformation for proportions. See Miller (1978).

- transf.ipft.hm: inverse of the Freeman-Tukey (double arcsine) transformation for proportions using the harmonic mean of the sample sizes for the back-transformation. See Miller (1978). The sample sizes are specified via the targs argument (the list element should be called ni).

- transf.isqrt: inverse of the square-root transformation (i.e., function to square a number).

- transf.irft: Freeman-Tukey transformation for incidence rates. See Freeman & Tukey (1950). The xi argument is used to specify the incidence rates and the ti argument the corresponding person-times at risk.

- transf.iirft: inverse of the Freeman-Tukey transformation for incidence rates.

- transf.ahw: Transformation of coefficient alpha as suggested by Hakstian & Whalen (1976).

- transf.iahw: Inverse of the transformation of coefficient alpha as suggested by Hakstian & Whalen (1976).

- transf.abt: Transformation of coefficient alpha as suggested by Bonett (2002).

- transf.iabt: Inverse of the transformation of coefficient alpha as suggested by Bonett (2002).

- transf.ztor.int: integral transformation method for the z-to-r transformation.

- transf.exp.int: integral transformation method for the exponential transformation.

- transf.ilogit.int: integral transformation method for the inverse of the logit transformation.

The integral transformation method for a transformation function $h(z)$ integrates $h(z)f(z)$ over $z$ using the limits targs$lower and targs$upper, where $f(z)$ is the density of a normal distribution with mean equal to xi and variance equal to targs$tau2. An example is provided below.

## Value

A vector with the transformed values.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Bonett, D. G. (2002). Sample size requirements for testing and estimating coefficient alpha. *Journal of Educational and Behavioral Statistics*, **27**, 335–340.

Fisher, R. A. (1921). On the "probable error" of a coefficient of correlation deduced from a small sample. *Metron*, **1**, 1–32.

Freeman, M. F., & Tukey, J. W. (1950). Transformations related to the angular and the square root. *Annals of Mathematical Statistics*, **21**, 607–611.

Hakstian, A. R., & Whalen, T. E. (1976). A k-sample significance test for independent alpha coefficients. *Psychometrika*, **41**, 219–231.

Miller, J. J. (1978). The inverse of the Freeman-Tukey double arcsine transformation. *American Statistician*, **32**, 138.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### average relative risk with 95% CI
predict(res, transf=exp)

### average relative risk with 95% CI using the integral transformation
predict(res, transf=transf.exp.int, targs=list(tau2=res$tau2, lower=-4, upper=4))
```

---

trimfill                            *Trim and Fill Analysis for 'rma.uni' Objects*

---

## Description

Carry out a trim and fill analysis for objects of class `"rma.uni"`.

## Usage

```
trimfill(x, ...)

## S3 method for class 'rma.uni'
trimfill(x, side, estimator="L0", maxiter=100, verbose=FALSE, ...)
```

### Arguments

| | |
|---|---|
| x | an object of class `"rma.uni"`. |
| side | either `"left"` or `"right"`, indicating on which side of the funnel plot the missing studies should be imputed. If left undefined, the side is chosen within the function depending on the results of Egger's regression test (see [regtest](#) for details on this test). |
| estimator | either `"L0"`, `"R0"`, or `"Q0"`, indicating the estimator to use for estimating the number of missing studies (default is `"L0"`). |
| maxiter | integer indicating the maximum number of iterations to use for the trim and fill method (default is `100`). |
| verbose | logical indicating whether output should be generated on the progress of the iterative algorithm used as part of the trim and fill method (default is `FALSE`). |
| ... | other arguments. |

### Details

The trim and fill method is a nonparametric (rank-based) data augmentation technique proposed by Duval and Tweedie (2000a, 2000b; see also Duval, 2005). The method can be used to estimate the number of studies missing from a meta-analysis due to the suppression of the most extreme results on one side of the funnel plot. The method then augments the observed data so that the funnel plot is more symmetric and recomputes the summary estimate based on the complete data. The trim and fill method can only be used in the context of a fixed- or random-effects model (i.e., in models without moderators). The method should not be regarded as a way of yielding a more "valid" estimate of the overall effect or outcome, but as a way of examining the sensitivity of the results to one particular selection mechanism (i.e., one particular form of publication bias).

### Value

An object of class `c("rma.uni.trimfill","rma.uni","rma")`. The object is a list containing the same components as objects created by [rma.uni](#), except that the data are augmented by the trim and fill method. The following components are also added:

| | |
|---|---|
| k0 | estimated number of missing studies. |
| side | either `"left"` or `"right"`, indicating on which side of the funnel plot the missing studies (if any) were imputed. |
| se.k0 | estimated standard error of k0. |
| p.k0 | p-value for the test of H_0: no missing studies on the chosen side (only when `estimator="R0"`; NA otherwise). |
| fill | a vector of dummy codes, indicating which of the data are the observed (0) and the augmented (1) data. |

The results of the fitted model after the data augmentation are printed with the [print.rma.uni](#) function. Calling [funnel.rma](#) on the object provides a funnel plot of the observed and augmented data.

**Note**

Three different estimators for the number of missing studies were proposed by Duval and Tweedie (2000a, 2000b). Based on these articles and Duval (2005), ″R0″ and ″L0″ are recommended. An advantage of estimator ″R0″ is that it provides a test of the null hypothesis that the number of missing studies (on the chosen side) is zero.

**Author(s)**

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

**References**

Duval, S. J., & Tweedie, R. L. (2000a). Trim and fill: A simple funnel-plot-based method of testing and adjusting for publication bias in meta-analysis. *Biometrics*, **56**, 455–463.

Duval, S. J., & Tweedie, R. L. (2000b). A nonparametric "trim and fill" method of accounting for publication bias in meta-analysis. *Journal of the American Statistical Association*, **95**, 89–98.

Duval, S. J. (2005). The trim and fill method. In H. R. Rothstein, A. J. Sutton, & M. Borenstein (Eds.) *Publication bias in meta-analysis: Prevention, assessment, and adjustments* (pp. 127–144). Chichester, England: Wiley.

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

**See Also**

funnel.rma

**Examples**

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a fixed-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg, method="FE")
res.tf <- trimfill(res)
res.tf
funnel(res.tf)

### estimator "R0" also provides test
res.tf <- trimfill(res, estimator="R0")
res.tf

### meta-analysis of the log relative risks using a random-effects model
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)
res.tf <- trimfill(res)
res.tf
funnel(res.tf)
```

---

update.rma | *Model Updating for 'rma' Objects*

---

### Description

The function is used to update and (by default) re-fit `"rma"` models. It does this by extracting the call stored in the object, updating the call and (by default) evaluating that call.

### Usage

```
## S3 method for class 'rma'
update(object, formula., ..., evaluate = TRUE)
```

### Arguments

object         an object of class `"rma"`.

formula.        changes to the formula. See 'Details'.

...              additional arguments to the call, or arguments with changed values.

evaluate        logical indicating whether to evaluate the new call or just return the call.

### Details

For objects of class `"rma.uni"`, `"rma.glmm"`, and `"rma.mv"`, the `formula.` argument can be used to update the set of moderators included in the model (see 'Examples').

### Value

If `evaluate=TRUE` the fitted object, otherwise the updated call.

### Author(s)

The present function is based on [update.default](), with changes made by Wolfgang Viechtbauer (<wvb@metafor-project.org>) so that the formula updating works with the (somewhat non-standard) interface of the [rma.uni](), [rma.glmm](), and [rma.mv]() functions.

### References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. <http://www.jstatsoft.org/v36/i03/>.

### See Also

[rma.uni](), [rma.mh](), [rma.peto](), [rma.glmm](), [rma.mv]()

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### calculate log relative risks and corresponding sampling variances
dat <- escalc(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg)

### random-effects model (method="REML" is default)
res <- rma(yi, vi, data=dat, digits=3)
res

### mixed-effects model with two moderators (absolute latitude and publication year)
res <- update(res, ~ ablat + year)
res

### remove 'year' moderator
res <- update(res, ~ . - year)
res

### fit model with ML estimation
update(res, method="ML")

### example with rma.glmm()
res <- rma.glmm(measure="OR", ai=tpos, bi=tneg, ci=cpos, di=cneg, data=dat.bcg, digits=3)
res <- update(res, mods = ~ ablat)
res

### conditional model with approximate likelihood
update(res, model="CM.AL")
```

---

vcov.rma                          *Extract the Variance-Covariance Matrix of the Parameter Estimates*
                                  *from 'rma' Objects*

---

## Description

The function extracts the (estimated) variance-covariance matrix of the (fixed effects) parameter estimates from objects of class "rma" or the marginal variance-covariance matrix of the observed effect sizes or outcomes.

## Usage

```
## S3 method for class 'rma'
vcov(object, type="fixed", ...)
```

## Arguments

object          an object of class "rma".

| type | character string indicating whether to return the variance-covariance matrix of the fixed effects (type=″fixed″) or the marginal variance-covariance matrix of the observed effect sizes or outcomes (type=″obs″). |
| ... | other arguments. |

## Details

Note that type=″obs″ currently only works for object of class ″rma.uni″ and ″rma.mv″.

## Value

A matrix corresponding to the requested variance-covariance matrix.

## Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

## References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

## See Also

rma.uni, rma.mh, rma.peto, rma.glmm, rma.mv

## Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
vcov(res)

### marginal var-cov matrix of the observed log relative risks
vcov(res, type="obs")
```

| weights.rma.uni | *Compute Weights for 'rma' Objects* |
|---|---|

### Description

The function computes the weights given to the observed effects or outcomes during the model fitting for objects of class "rma.uni", "rma.mh", "rma.peto", and "rma.mv".

### Usage

```
## S3 method for class 'rma.uni'
weights(object, type="diagonal", ...)
## S3 method for class 'rma.mh'
weights(object, type="diagonal", ...)
## S3 method for class 'rma.peto'
weights(object, type="diagonal", ...)
## S3 method for class 'rma.glmm'
weights(object, ...)
## S3 method for class 'rma.mv'
weights(object, type="diagonal", ...)
```

### Arguments

| | |
|---|---|
| object | an object of class "rma.uni", "rma.mh", "rma.peto", or "rma.mv". The method is not yet implemented for objects of class "rma.glmm". |
| type | character string indicating whether to only return the diagonal of the weight matrix ("diagonal") or the entire matrix ("matrix"). |
| ... | other arguments. |

### Value

Either a vector with the diagonal elements of the weight matrix or the entire weight matrix. When only the diagonal elements are returned, they are given in % (and they add up to 100%). When the entire weight matrix is requested, this is always a diagonal matrix for objects of class "rma.uni", "rma.mh", "rma.peto".

### Author(s)

Wolfgang Viechtbauer <wvb@metafor-project.org>
package website: http://www.metafor-project.org/
author homepage: http://www.wvbauer.com/

### References

Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, **36**(3), 1–48. http://www.jstatsoft.org/v36/i03/.

### See Also

rma.uni, rma.mh, rma.peto, rma.mv, influence.rma.uni

### Examples

```
### load BCG vaccine data
data(dat.bcg)

### meta-analysis of the log relative risks using a mixed-effects model
### with two moderators (absolute latitude and publication year)
res <- rma(measure="RR", ai=tpos, bi=tneg, ci=cpos, di=cneg,
           mods = ~ ablat + year, data=dat.bcg)
weights(res)
```

# Index