

Journal of Statistical Software

August 2010, Volume 36, Issue 3. *http://www.jstatsoft.org/*

Conducting Meta-Analyses in R with the metafor Package

Wolfgang Viechtbauer
Maastricht University

Abstract

The **metafor** package provides functions for conducting meta-analyses in R. The package includes functions for fitting the meta-analytic fixed- and random-effects models and allows for the inclusion of moderators variables (study-level covariates) in these models. Meta-regression analyses with continuous and categorical moderators can be conducted in this way. Functions for the Mantel-Haenszel and Peto's one-step method for meta-analyses of 2×2 table data are also available. Finally, the package provides various plot functions (for example, for forest, funnel, and radial plots) and functions for assessing the model fit, for obtaining case diagnostics, and for tests of publication bias.

Keywords: meta-analysis, R, mixed-effects model, meta-regression, moderator analysis.

1. Introduction

Science is a cumulative process (Shoemaker *et al.* 2003). Therefore, it is not surprising that one can often find dozens and sometimes hundreds of studies addressing the same basic question. Researches trying to aggregate and synthesize the literature on a particular topic are increasingly conducting meta-analyses (Olkin 1995). Broadly speaking, a meta-analysis can be defined as a systematic literature review supported by statistical methods where the goal is to aggregate and contrast the findings from several related studies (Glass 1976).

In a meta-analysis, the relevant results from each study are quantified in such a way that the resulting values can be further aggregated and compared. For example, we may be able to express the results from a randomized clinical trial examining the effectiveness of a medication in terms of an odds ratio, indicating how much higher/lower the odds of a particular outcome (e.g., remission) were in the treatment compared to the control group. The set of odds ratios from several studies examining the same medication then forms the data which is used for further analyses. For example, we can estimate the average effectiveness of the medication

(i.e., the average odds ratio) or conduct a moderator analysis, that is, we can examine whether the effectiveness of the medication depends on the characteristics of the studies (e.g., the size of the odds ratio may depend on the medication dosage used in the various trials).

Meta-analyses can also be used to aggregate estimates of the strength of the relationship between two variables measured concurrently and/or without manipulation by experimenters (e.g., gender differences in risk taking behaviors, the relationship between job satisfaction and job performance). Again, the idea is that the relevant results of each study are expressed in terms of an outcome measure putting the results on a common scale. Depending on the types of studies and the information provided therein, a variety of different outcome measures can be used for a meta-analysis, including the odds ratio, relative risk, risk difference, the correlation coefficient, and the (standardized) mean difference (e.g., Borenstein 2009; Fleiss and Berlin 2009). The term "effect size" is used generically throughout this article to denote the outcome measure chosen for a meta-analysis (and does not imply that we are necessarily dealing with a measure that indicates the causal influence of one variable on another).

Several standalone software packages dedicated specifically for conducting meta-analyses have been made available over the recent years. Commercial packages include **MetaWin** (Rosenberg *et al.* 2000) and **Comprehensive Meta-Analysis** (Borenstein *et al.* 2005). The freely available **RevMan** (Review Manager) from The Cochrane Collaboration (2008) not only provides functions for conducting meta-analyses, but actually comprises an entire toolset for preparing and maintaining Cochrane reviews.

Existing software packages have also been extended to provide meta-analytic capabilities. **MIX** (Bax *et al.* 2006) and **MetaEasy** (Kontopantelis and Reeves 2009) are add-ins for **Excel** and meta-analysis functions/macros have been made available for **Stata** (StataCorp. 2007; for details, see Sterne 2009) and SPSS (SPSS Inc. 2006; for details, see Lipsey and Wilson 2001). Using the `proc mixed` command, one can also carry out meta-analyses using SAS (SAS Institute Inc. 2003; for details, see van Houwelingen *et al.* 2002). Several meta-analysis packages are also available for R (R Development Core Team 2010), e.g., **meta** (Schwarzer 2010) and **rmeta** (Lumley 2009).

The existing R packages, however, currently only provide limited capabilities for conducting moderator analyses (Section 5 provides a more detailed comparison between various packages). The **metafor** package (Viechtbauer 2010), which is described in the present paper, provides functions for conducting meta-analyses in R and includes the required methods for conducting moderator analyses without such limitations. In particular, users can fit so-called meta-regression models (e.g., Berkey *et al.* 1995; van Houwelingen *et al.* 2002), that is, linear models that examine the influence of one or more moderator variables on the outcomes. With appropriate coding, such models can handle continuous and categorical moderator variables.

The **metafor** package grew out of a function called `mima()` (Viechtbauer 2006), which was written by the author several years ago and which has since been successfully applied in several meta-analyses (e.g., Krasopoulos *et al.* 2008; Petrin *et al.* 2008; Roberts *et al.* 2006). While the `mima()` function provided the basic functionality for fitting standard meta-analytic models and conducting meta-regression analyses, the **metafor** package was written in response to several requests to expand the `mima()` function into a full package for conducting meta-analyses with additional options and support functions.

In particular, the **metafor** package currently includes functions for fitting the meta-analytic fixed- and random-effects models and allows for the inclusion of moderator variables in these

models. Functions for the Mantel-Haenszel and Peto's one-step method are also available. Finally, the package provides various plot functions (for example, for forest, funnel, and radial plots) and functions for assessing the model fit, for obtaining case diagnostics, and for tests of publication bias.

The purpose of the present article is to provide a general overview of the **metafor** package and its current capabilities. Not all of the possibilities and options are described, as this would require a much longer treatment. The article is therefore a starting point for those interested in exploring the possibility of conducting meta-analyses in R with the **metafor** package. Plans for extending the package are described at the end of the article.

2. Meta-analysis models

In this section, the meta-analytic fixed- and random/mixed-effects models are briefly described (e.g., Hedges and Olkin 1985; Berkey *et al.* 1995; van Houwelingen *et al.* 2002; Raudenbush 2009). These models form the basis for most meta-analyses and are also the models underlying the **metafor** package. We start with $i = 1, \ldots, k$ independent effect size estimates, each estimating a corresponding (true) effect size. We assume that

$$y_i = \theta_i + e_i, \tag{1}$$

where y_i denotes the observed effect in the i-th study, θ_i the corresponding (unknown) true effect, e_i is the sampling error, and $e_i \sim N(0, v_i)$. Therefore, the y_i's are assumed to be unbiased and normally distributed estimates of their corresponding true effects. The sampling variances (i.e., 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, Fisher's r-to-z transformation for correlations; see Section 3.1 for more details).

2.1. Random-effects model

Most meta-analyses are based on sets of studies that are not exactly identical in their methods and/or the characteristics of the included samples. Differences in the methods and sample characteristics may introduce variability ("heterogeneity") among the true effects. One way to model the heterogeneity is to treat it as purely random. This leads to the random-effects model, given by

$$\theta_i = \mu + u_i, \tag{2}$$

where $u_i \sim N(0, \tau^2)$. Therefore, the true effects are assumed to be normally distributed with mean μ and variance τ^2. The goal is then to estimate μ, the average true effect and τ^2, the (total) amount of heterogeneity among the true effects. If $\tau^2 = 0$, then this implies homogeneity among the true effects (i.e., $\theta_1 = \ldots = \theta_k \equiv \theta$), so that $\mu = \theta$ then denotes *the* true effect.

2.2. Mixed-effects model

Alternatively, we can include one or more moderators (study-level variables) in the model that may account for at least part of the heterogeneity in the true effects. This leads to the

mixed-effects model, given by

$$\theta_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_{p'} x_{ip'} + u_i, \tag{3}$$

where x_{ij} denotes the value of the j-th moderator variable for the i-th study and we assume again that $u_i \sim N(0, \tau^2)$. Here, τ^2 denotes the amount of *residual* heterogeneity among the true effects, that is, variability among the true effects that is not accounted for by the moderators included in the model. The goal of the analysis is then to examine to what extent the moderators included in the model influence the size of the average true effect.

2.3. Fixed-effects vs. random/mixed-effects models

The random/mixed-effects models need to be carefully distinguished from fixed-effects models. When using fixed-effects models, the goal is to make a *conditional inference* only about the k studies included in the meta-analysis (Hedges and Vevea 1998). For example, a fixed-effects model without moderators provides an answer to the question: How large is the average true effect in the set of k studies included in the meta-analysis?

To be precise, the question addressed by a fixed-effects model depends on the type of estimation method used. If weighted least squares is used to fit the model, then the fixed-effects model provides an estimate of

$$\bar{\theta}_w = \frac{\sum w_i \theta_i}{\sum w_i}, \tag{4}$$

the weighted average of the true effects, where the weights are typically set equal to $w_i = 1/v_i$. On the other hand, unweighted least squares provides an estimate of

$$\bar{\theta}_u = \frac{\sum \theta_i}{k}, \tag{5}$$

the simple (unweighted) average of the true effects (Laird and Mosteller 1990).

In contrast to the fixed-effects model, random/mixed-effects models provide an *unconditional inference* about a larger set of studies from which the k studies included in the meta-analysis are assumed to be a random sample (Hedges and Vevea 1998). We typically do not assume that this larger set consists only of studies that have actually been conducted, but instead envision a hypothetical population of studies that comprises studies that have been conducted, that could have been conducted, or that may be conducted in the future. The random-effects model then addresses the question: How large is the average true effect in this larger population of studies (i.e., how large is μ)?

Therefore, 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 are homogeneous. In other words, fixed-effects models provide perfectly valid inferences under heterogeneity, as long as one is restricting these inferences (i.e., the conclusions about the size of the average effect) to the set of studies included in the meta-analysis.[1] On the other hand, the random-effects model provides an inference about the average effect in the entire population of studies from which the included studies are assumed to be a random selection.

In the special case that the true effects are actually homogeneous, the distinction between the various models disappears, since homogeneity implies that $\mu = \bar{\theta}_w = \bar{\theta}_u \equiv \theta$. However,

[1] More specifically, to sets of k studies with true effects equal to the true effects of the k studies included in the meta-analysis.

since there is no infallible method to test whether the true effects are really homogeneous or not, a researcher should decide on the type of inference desired before examining the data and choose the model accordingly. For more details on the distinction between fixed- and random-effects models, see Hedges and Vevea (1998) and Laird and Mosteller (1990).

2.4. Model fitting

In essence, the various meta-analytic models are just special cases of the general linear (mixed-effects) model with heteroscedastic sampling variances that are assumed to be known. The random/mixed-effects models can therefore be fitted using a two step approach (Raudenbush 2009). First, the amount of (residual) heterogeneity (i.e., τ^2) is estimated with one of the various estimators that have been suggested in the literature, including the Hunter-Schmidt estimator (Hunter and Schmidt 2004), the Hedges estimator (Hedges and Olkin 1985; Raudenbush 2009), the DerSimonian-Laird estimator (DerSimonian and Laird 1986; Raudenbush 2009), the Sidik-Jonkman estimator (Sidik and Jonkman 2005a,b), the maximum-likelihood or restricted maximum-likelihood estimator (Viechtbauer 2005; Raudenbush 2009), or the empirical Bayes estimator (Morris 1983; Berkey *et al.* 1995). Next, μ or $\beta_0, \ldots, \beta_{p'}$ are estimated via weighted least squares with weights equal to $w_i = 1/(v_i + \hat{\tau}^2)$, where $\hat{\tau}^2$ denotes the estimate of τ^2.

Once the parameter estimates have been obtained, Wald-type tests and confidence intervals (CIs) are then easily obtained for μ or $\beta_0, \ldots, \beta_{p'}$ under the assumption of normality. For models involving moderators, subsets of the parameters can also be tested in the same manner. Based on the fitted model, we can also obtain fitted/predicted values, residuals, and the best linear unbiased predictions (BLUPs) of the study-specific true effects. The null hypothesis $\mathrm{H}_0 : \tau^2 = 0$ in random- and mixed-effects models can be tested with Cochran's Q-test (Hedges and Olkin 1985). A confidence interval for τ^2 can be obtained with the method described in Viechtbauer (2007a).

Fixed-effects models can be fitted with either weighted or unweighted least squares, again taking into consideration the heteroscedastic sampling variances.[2] As mentioned above, this provides either an estimate of the weighted or unweighted average of the true effects when not including moderators in the model. With moderators in the model, weighted estimation provides an estimate of the weighted least squares relationship between the moderator variables and the true effects, while unweighted estimation provides an estimate of the unweighted least squares relationship.

3. The metafor package

The **metafor** package provides functions for fitting the various models described above. The package is available via the Comprehensive R Archive Network (CRAN) at `http://CRAN.R-project.org/package=metafor`, the author's website at `http://www.wvbauer.com/`, or can be directly installed within R by typing `install.packages("metafor")` (assuming an internet connection and appropriate access rights on the computer). The current version number is 1.4-0. Once the package has been installed, it should be possible to replicate the

[2]In principle, one can also choose between weighted and unweighted least squared when fitting random/mixed-effects models. However, since the parameters remain the same regardless of the method used, weighted estimation is usually to be preferred since it is more efficient.

analyses described in this paper.

3.1. Calculating outcome measures

Before beginning with a meta-analysis, one must first obtain a set of effect size estimates with their corresponding sampling variances. If these have been calculated already, for example by hand or with the help of other software, then these can be read-in from a text file with the `read.table()` function (see `help("read.table")` for details).

The **metafor** package also provides the `escalc()` function, which can be used to calculate various effect size or outcome measures (and the corresponding sampling variances) that are commonly used in meta-analyses. There are two different interfaces for using this function, a default and a formula interface. For the default interface, the arguments of the function are

```
escalc(measure, ai, bi, ci, di, n1i, n2i, m1i, m2i, sd1i, sd2i, xi, mi, ri,
  ni, data = NULL, add = 1/2, to = "only0", vtype = "LS", append = FALSE)
```

where `measure` is a character string specifying which outcome measure should be calculated (see below for the various options), arguments `ai` through `ni` are used to supply the needed information to calculate the various measures (depending on the outcome measure specified under `measure`, different arguments need to be supplied), `data` can be used to specify a data frame containing the variables given to the previous arguments, `add` and `to` are arguments needed when dealing with 2×2 table data that may contain cells with zeros, and `vtype` is an argument specifying the sampling variance estimate that should be calculated (see below). When setting `append = TRUE`, the data frame specified via the `data` argument is returned together with the effect size estimates and corresponding sampling variances.

Outcome measures for 2×2 *table data*

Meta-analyses in the health/medical sciences are often based on studies providing data in terms of 2×2 tables. In particular, assume that we have k tables of the form:

	outcome 1	outcome 2	
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 in the i-th study. For example, in a set of randomized clinical trials, group 1 and group 2 may refer to the treatment and placebo/control group, with outcome 1 denoting some event of interest (e.g., remission) 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. The 2×2 tables may also be the result of cross-sectional (i.e., multinomial) sampling, so that none of the table margins (except the total sample size `n1i + n2i`) are fixed by the study design.

Depending on the type of design (sampling method), a meta-analysis of 2×2 table data can be based on one of several different outcome measures, including the odds ratio, the relative

risk (also called risk ratio), the risk difference, and the arcsine transformed risk difference (e.g., Fleiss and Berlin 2009). For these measures, one needs to supply either `ai`, `bi`, `ci`, and `di` or alternatively `ai`, `ci`, `n1i`, and `n2i`. The options for the `measure` argument are then:

- `"RR"`: The log relative risk is equal to the log of `(ai/n1i)/(ci/n2i)`.
- `"OR"`: The log odds ratio is equal to the log of `(ai*di)/(bi*ci)`.
- `"RD"`: The risk difference is equal to `(ai/n1i) - (ci/n2i)`.
- `"AS"`: The arcsine transformed risk difference is equal to `asin(sqrt(ai/n1i)) - asin(sqrt(ci/n2i))`. See Rücker *et al.* (2009) for a discussion of this and other outcome measures for 2×2 table data.
- `"PETO"`: The log odds ratio estimated with Peto's method (Yusuf *et al.* 1985) is equal to `(ai - si * n1i/ni)/((si * ti * n1i * n2i)/(ni^2 * (ni - 1)))`, where `si = ai + ci`, `ti = bi + di`, and `ni = n1i + n2i`. This measure technically assumes that the true odds ratio is equal to 1 in all tables.

Note that the log is taken of the relative risk and the odds ratio, which makes these outcome measures symmetric around 0 and helps to make the distribution of these outcome measure closer to normal.

Cell entries with a zero can be problematic especially for the relative risk and the odds ratio. Adding a small constant to the cells of the 2×2 tables is a common solution to this problem. When `to = "all"`, the value of `add` is added to each cell of the 2×2 tables in all k tables. When `to = "only0"`, the value of `add` is added to each cell of the 2×2 tables only in those tables with at least one cell equal to 0. When `to = "if0all"`, the value of `add` is added to each cell of the 2×2 tables in all k tables, but only when there is at least one 2×2 table with a zero entry. 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 presence of zero cells, this may lead to division by zero inside of the function (when this occurs, the resulting `Inf` values are recoded to `NA`).

Raw and standardized mean differences

The raw mean difference and the standardized mean difference are useful effect size measures when meta-analyzing a set of studies comparing two experimental groups (e.g., treatment and control groups) or two naturally occurring groups (e.g., men and women) with respect to some quantitative (and ideally normally distributed) dependent variable (e.g., Borenstein 2009). For these outcome measures, `m1i` and `m2i` are used to specify the means of the two groups, `sd1i` and `sd2i` the standard deviations of the scores in the two groups, and `n1i` and `n2i` the sample sizes of the two groups.

- `"MD"`: The raw mean difference is equal to `m1i - m2i`.
- `"SMD"`: The standardized mean difference is equal to `(m1i - m2i)/spi`, where `spi` is the pooled standard deviation of the two groups (which is calculated inside of the function). The standardized mean difference is automatically corrected for its slight positive bias within the function (Hedges and Olkin 1985). When `vtype = "LS"`, the

sampling variances are calculated based on a large sample approximation. Alternatively, the unbiased estimates of the sampling variances can be obtained with `vtype = "UB"`.

Raw and transformed correlation coefficients

Another frequently used outcome measure for meta-analyses is the correlation coefficient, which is used to measure the strength of the (linear) relationship between two quantitative variables (e.g., Borenstein 2009). Here, one needs to specify `ri`, the vector with the raw correlation coefficients, and `ni`, the corresponding sample sizes.

- `"COR"`: The raw correlation coefficient is simply equal to `ri` as supplied to the function. When `vtype = "LS"`, the sampling variances are calculated based on the large sample approximation. Alternatively, an approximation to the unbiased estimates of the sampling variances can be obtained with `vtype = "UB"` (Hedges 1989).
- `"UCOR"`: The unbiased estimate of the correlation coefficient is obtained by correcting the raw correlation coefficient for its slight negative bias (based on equation 2.7 in Olkin and Pratt 1958). Again, `vtype = "LS"` and `vtype = "UB"` can be used to choose between the large sample approximation or approximately unbiased estimates of the sampling variances.
- `"ZCOR"`: Fisher's r-to-z transformation is a variance stabilizing transformation for correlation coefficients with the added benefit of also being a rather effective normalizing transformation (Fisher 1921). The Fisher's r-to-z transformed correlation coefficient is equal to `1/2 * log((1 + ri)/(1 - ri))`.

Proportions and transformations thereof

When the studies provide data for single groups with respect to a dichotomous dependent variable, then the raw proportion, the logit transformed proportion, the arcsine transformed proportion, and the Freeman-Tukey double arcsine transformed proportion are useful outcome measures. 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.

- `"PR"`: The raw proportion is equal to `xi/ni`.
- `"PLO"`: The logit transformed proportion is equal to the log of `xi/(ni - xi)`.
- `"PAS"`: The arcsine transformation is a variance stabilizing transformation for proportions and is equal to `asin(sqrt(xi/ni))`.
- `"PFT"`: Yet another variance stabilizing transformation for proportions was suggested by Freeman and Tukey (1950). The Freeman-Tukey double arcsine transformed proportion is equal to `1/2 * (asin(sqrt(xi/(ni + 1))) + asin(sqrt((xi + 1)/(ni + 1))))`.

Again, zero cell entries can be problematic. When `to = "all"`, the value of `add` is added to `xi` and `mi` in all k studies. When `to = "only0"`, the value of `add` is added only for studies where `xi` or `mi` is equal to 0. When `to = "if0all"`, the value of `add` is added in all k studies, but only when there is at least one study with a zero value for `xi` or `mi`. Setting `to = "none"` or `add = 0` again means that no adjustment to the observed values is made.

Formula interface

The `escalc()` function also provides an alternative formula interface to specify the data structure. The arguments for this interface are

```
escalc(measure, formula, weights, data,
  add = 1/2, to = "only0", vtype = "LS")
```

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 (based on the **Formula** package; see Zeileis and Croissant 2010) together with the `weights` argument for the group sizes or cell frequencies. The `data` argument can be used to specify a data frame containing the variables in the formula and the `weights` variable. The `add`, `to`, and `vtype` arguments work as described above.

For 2×2 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, and `study` is a k-level factor specifying the studies. The `weights` argument is then used to specify the frequencies in the various cells. An example to illustrate the default and the formula interface is given in the following section.

3.2. Example

The **metafor** package provides the data set object `dat.bcg` with the results from 13 studies on the effectiveness of the BCG vaccine against tuberculosis (Colditz *et al.* 1994).

```
R> library("metafor")
R> data("dat.bcg", package = "metafor")
R> print(dat.bcg, row.names = FALSE)

trial               author year tpos  tneg cpos  cneg ablat       alloc
    1              Aronson 1948    4   119   11   128    44      random
    2     Ferguson & Simes 1949    6   300   29   274    55      random
    3      Rosenthal et al 1960    3   228   11   209    42      random
    4    Hart & Sutherland 1977   62 13536  248 12619    52      random
    5 Frimodt-Moller et al 1973   33  5036   47  5761    13  alternate
    6      Stein & Aronson 1953  180  1361  372  1079    44  alternate
    7     Vandiviere et al 1973    8  2537   10   619    19      random
    8           TPT Madras 1980  505 87886  499 87892    13      random
    9     Coetzee & Berjak 1968   29  7470   45  7232    27      random
   10      Rosenthal et al 1961   17  1699   65  1600    42 systematic
   11       Comstock et al 1974  186 50448  141 27197    18 systematic
   12   Comstock & Webster 1969    5  2493    3  2338    33 systematic
   13       Comstock et al 1976   27 16886   29 17825    33 systematic
```

Besides the trial number, author(s), and publication year, the data set includes information about the number of treated (vaccinated) subjects that were tuberculosis positive and negative (`tpos` and `tneg`, respectively) and similarly for the control (non-vaccinated) subjects (`cpos` and `cneg`, respectively). In addition, the absolute latitude of the study location (in degrees) and the treatment allocation method (random, alternate, or systematic assignment) are indicated for each trial.

The results of the studies can be expressed in terms of 2×2 tables, given by

	TB+	TB-
Treated	`tpos`	`tneg`
Control	`cpos`	`cneg`

for which one of the previously mentioned outcome measures can be calculated. In the following examples, we will work with the (log) relative risk as the outcome measure. We can obtain these values and corresponding sampling variances with:

```
R> dat <- escalc(measure = "RR", ai = tpos, bi = tneg, ci = cpos,
+    di = cneg, data = dat.bcg, append = TRUE)
R> print(dat[,-c(4:7)], row.names = FALSE)
```

```
trial               author year ablat      alloc          yi          vi
    1              Aronson 1948    44     random -0.88931133 0.325584765
    2     Ferguson & Simes 1949    55     random -1.58538866 0.194581121
    3      Rosenthal et al 1960    42     random -1.34807315 0.415367965
    4    Hart & Sutherland 1977    52     random -1.44155119 0.020010032
    5 Frimodt-Moller et al 1973    13  alternate -0.21754732 0.051210172
    6      Stein & Aronson 1953    44  alternate -0.78611559 0.006905618
    7     Vandiviere et al 1973    19     random -1.62089822 0.223017248
    8           TPT Madras 1980    13     random  0.01195233 0.003961579
    9     Coetzee & Berjak 1968    27     random -0.46941765 0.056434210
   10      Rosenthal et al 1961    42 systematic -1.37134480 0.073024794
   11       Comstock et al 1974    18 systematic -0.33935883 0.012412214
   12   Comstock & Webster 1969    33 systematic  0.44591340 0.532505845
   13       Comstock et al 1976    33 systematic -0.01731395 0.071404660
```

For interpretation purposes, it is important to note that the log relative risks were calculated in such a way that values below 0 indicate a *lower* infection risk for the vaccinated group. Except for two cases, this is also the direction of the findings in these 13 studies.

To use the formula interface of the `escalc()` function, we must first rearrange the data into the required (long) format:

```
R> k <- length(dat.bcg$trial)
R> dat.fm <- data.frame(study = factor(rep(1:k, each = 4)))
R> dat.fm$grp <- factor(rep(c("T", "T", "C", "C"), k), levels = c("T", "C"))
```

```
R> dat.fm$out <- factor(rep(c("+", "-", "+", "-"), k), levels = c("+", "-"))
R> dat.fm$freq <- with(dat.bcg, c(rbind(tpos, tneg, cpos, cneg)))
R> dat.fm
```

with the first eight rows of the rearranged data shown below.

```
   study grp out  freq
1      1   T   +     4
2      1   T   -   119
3      1   C   +    11
4      1   C   -   128
5      2   T   +     6
6      2   T   -   300
7      2   C   +    29
8      2   C   -   274
...
```

With

```
R> escalc(out ~ grp | study, weights = freq, data = dat.fm, measure = "RR")
```

we then obtain

```
            yi          vi
1  -0.88931133 0.325584765
2  -1.58538866 0.194581121
3  -1.34807315 0.415367965
4  -1.44155119 0.020010032
.           .           .
13 -0.01731395 0.071404660
```

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.

3.3. Fitting models

The various meta-analytic models can be fitted with the `rma.uni()` function (with alias `rma()`). The models are fitted as described in Section 2.4. The arguments of the function are given by

```
rma.uni(yi, vi, sei, ai, bi, ci, di, n1i, n2i, m1i, m2i, sd1i, sd2i, xi,
  mi, ri, ni, mods = NULL, data = NULL, intercept = TRUE, slab = NULL,
  subset = NULL, measure = "GEN", add = 1/2, to = "only0", vtype = "LS",
  method = "REML", weighted = TRUE, level = 95, digits = 4, btt = NULL,
  tau2 = NULL, knha = FALSE, control = list())
```

which are explained below.

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, standardized mean differences, correlation coefficients). One simply needs to supply the observed outcomes via the `yi` argument and the corresponding sampling variances via the `vi` argument (or the standard errors, the square root of the sampling variances, via the `sei` argument). When specifying the data in this way, one must set `measure = "GEN"` (which is the default).

Alternatively, the function takes as input the same arguments as the `escalc()` function and then automatically calculates the values for the chosen effect size or outcome measure (and the corresponding sampling variances) when supplied with the needed data. The `measure` argument is then used to specify the desired outcome measure (arguments `add`, `to`, and `vtype` have the same meaning as for the `escalc()` function).

Specifying the model

Assuming the observed outcomes and corresponding sampling variances are supplied via `yi` and `vi`, the random-effects model is fitted with `rma(yi, vi, data = dat)`. Restricted maximum-likelihood estimation is used by default when estimating τ^2 (the REML estimator is approximately unbiased and quite efficient; see Viechtbauer 2005). The various (residual) heterogeneity estimators that can be specified via the `method` argument are the

- `"HS"`: Hunter-Schmidt estimator.
- `"HE"`: Hedges estimator.
- `"DL"`: DerSimonian-Laird estimator.
- `"SJ"`: Sidik-Jonkman estimator.
- `"ML"`: Maximum-likelihood estimator.
- `"REML"`: Restricted maximum-likelihood estimator.
- `"EB"`: Empirical Bayes estimator.

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 design matrix with k rows and p' columns (e.g., using `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 included in the model by default unless the user sets `intercept = FALSE`.

Many R user will be familiar with the formula syntax used to specify the desired model in functions such as `lm()` and `glm()` (see `help("formula")` for details). One can also specify the desired meta-analytic model in this way by setting the `mods` argument equal to a one-sided formula of the form `~ 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 remove the intercept term).

A fixed-effects model can be fitted with `rma(yi, vi, data = dat, method = "FE")`. Here, one must consider carefully whether weighted or unweighed least squares should be used (the default is `weighted = TRUE`). Again, moderators can be included in the model via the `mods` argument.

Omnibus test of parameters

For models including moderators, an omnibus test of all the model coefficients is conducted that excludes the intercept β_0 (the first coefficient) if it is included in the model (i.e., a test of $\mathrm{H}_0 : \beta_1 = \ldots = \beta_{p'} = 0$). 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, `btt = c(3, 4)` would be used to include only the third and fourth coefficient from the model in the test (if an intercept is included in the model, then it corresponds to the first coefficient in the model).

Categorical moderator variables

Categorical moderator variables can be included in the model in the same way that appropriately (dummy) coded categorical independent variables can be included in linear models in general. 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. An example to illustrate these different approaches is provided below.

Knapp and 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 χ^2 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 τ^2 and leads to different reference distributions. 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 adjustment is only meant to be used in the context of random- or mixed-effects model.

3.4. Example

Random-effects model

We will now start by fitting a random-effects model to the BCG data. Both of the commands

```
R> res <- rma(yi, vi, data = dat)
R> res
```

and

```
R> res <- rma(ai = tpos, bi = tneg, ci = cpos, di = cneg, data = dat,
+    measure = "RR")
R> res
```

yield the same output, namely

```
Random-Effects Model (k = 13; tau^2 estimator: REML)

tau^2 (estimate of total amount of heterogeneity): 0.3132 (SE = 0.1664)
tau (sqrt of the estimate of total heterogeneity): 0.5597
I^2 (% of total variability due to heterogeneity): 92.22%
H^2 (total variability / within-study variance):   12.86

Test for Heterogeneity:
Q(df = 12) = 152.2330, p-val < .0001

Model Results:

estimate       se     zval     pval    ci.lb    ci.ub
 -0.7145   0.1798  -3.9744   <.0001  -1.0669  -0.3622      ***

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

indicating that the estimated average log relative risk is equal to $\hat{\mu} = -0.7145$ (95% CI: -1.0669 to -0.3622). For easier interpretation, it may be useful to transform these values back to the relative risk scale through exponentiation (i.e., $\exp(\hat{\mu}) = 0.49$ with 95% CI: 0.34 to 0.70). The results therefore suggest that the risk of a tuberculosis infection in vaccinated individuals is on average half as large as the infection risk without the vaccination. The null hypothesis $\mathrm{H}_0\colon \mu = 0$ can be clearly rejected ($z = -3.97$, $p < 0.0001$).

The amount of heterogeneity in the true log relative risks is estimated to be $\hat{\tau}^2 = 0.3132$. Various measures for facilitating the interpretation of the estimated amount of heterogeneity were suggested by Higgins and Thompson (2002). The I^2 statistic estimates (in percent) how much of the total variability in the effect size estimates (which is composed of heterogeneity and sampling variability) can be attributed to heterogeneity among the true effects ($\hat{\tau}^2 = 0$ therefore implies $I^2 = 0\%$). The H^2 statistic is the ratio of the total amount of variability in the observed outcomes to the amount of sampling variability ($\hat{\tau}^2 = 0$ therefore implies $H^2 = 1$). It is important to realize, however, that $\hat{\tau}^2$, I^2, and H^2 are often estimated imprecisely, especially when the number of studies is small. With

```
R> confint(res)
```

we can obtain corresponding confidence intervals

```
       estimate   ci.lb   ci.ub
tau^2    0.3132  0.1197  1.1115
tau      0.5597  0.3460  1.0543
I^2(%)  92.2214 81.9177 97.6781
H^2     12.8558  5.5303 43.0677
```

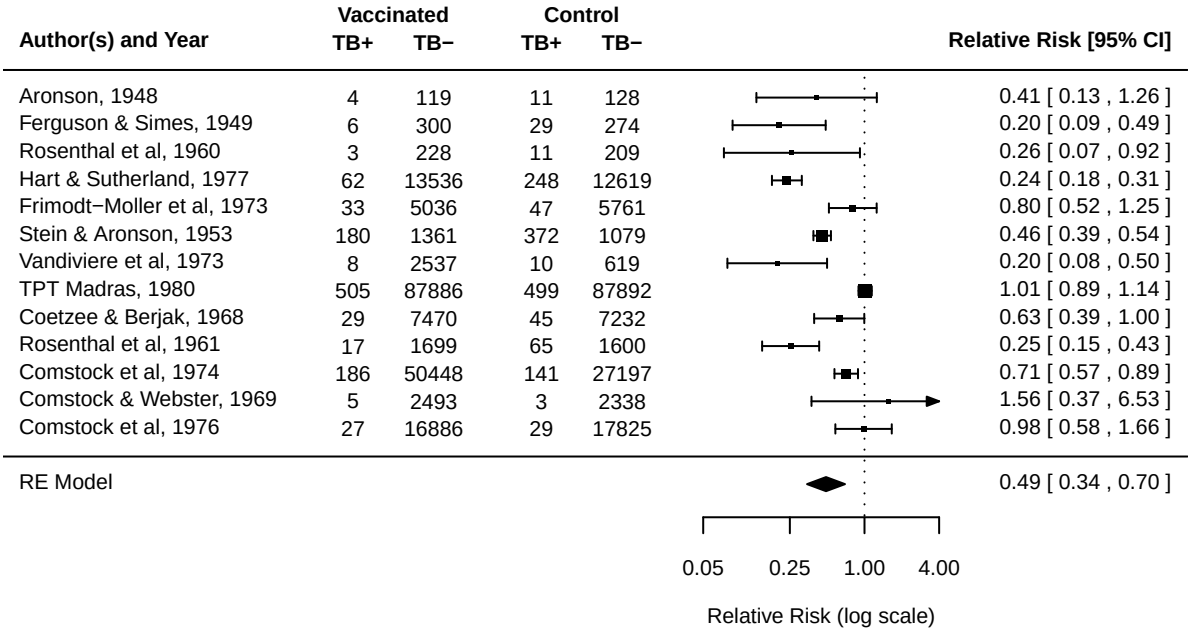


Figure 1: Forest plot showing the results of 13 studies examining the effectiveness of the BCG vaccine for preventing tuberculosis. The figure shows the relative risk of a tuberculosis infection in the treated versus the control group with corresponding 95% confidence intervals in the individual studies and based on a random-effects model.

which are quite wide and therefore indicate that we should not give too much credence to the exact point estimates. However, even the lower bound values of the confidence intervals are quite large and the test for heterogeneity ($Q = 152.23$, $df = 12$, $p < 0.0001$) suggests considerable heterogeneity among the true effects.

A graphical overview of the results so far can be obtained by creating a forest plot (Lewis and Clarke 2001) with the `forest()` function. While `forest(res)` would be sufficient, a more appealing figure can be produced with some extra code (see Figure 1). By default, the observed effects are drawn proportional to the precision of the estimates. The summary estimate based on the random-effects model is automatically added to the figure (with the outer edges of the polygon indicating the confidence interval limits). The results are shown using a log scale for easier interpretation. The figure was created with the following code.

```
R> forest(res, slab = paste(dat$author, dat$year, sep = ", "),
+    xlim = c(-16, 6), at = log(c(0.05, 0.25, 1, 4)), atransf = exp,
+    ilab = cbind(dat$tpos, dat$tneg, dat$cpos, dat$cneg),
+    ilab.xpos = c(-9.5, -8, -6, -4.5), cex = 0.75)
R> op <- par(cex = 0.75, font = 2)
R> text(c(-9.5, -8, -6, -4.5), 15, c("TB+", "TB-", "TB+", "TB-"))
R> text(c(-8.75, -5.25), 16, c("Vaccinated", "Control"))
R> text(-16, 15, "Author(s) and Year", pos = 4)
R> text(6, 15, "Relative Risk [95% CI]", pos = 2)
R> par(op)
```

Mixed-effects model

At least part of the heterogeneity may be due to the influence of moderators. For example, the effectiveness of the BCG vaccine may depend on the study location, as the increased abundance of non-pathogenic environmental mycobacteria closer to the equator may provide a natural protection from tuberculosis (Ginsberg 1998). Moreover, the effectiveness of the vaccine may have changed over time. We can examine these hypotheses by fitting a mixed-effects model including the absolute latitude and publication year of the studies as moderators.

```
R> res <- rma(yi, vi, mods = cbind(ablat, year), data = dat)
R> res
```

and

```
R> res <- rma(yi, vi, mods = ~ ablat + year, data = dat)
R> res
```

produce the same results, namely

```
Mixed-Effects Model (k = 13; tau^2 estimator: REML)

tau^2 (estimate of residual amount of heterogeneity): 0.1108 (SE = 0.0845)
tau (sqrt of the estimate of residual heterogeneity): 0.3328

Test for Residual Heterogeneity:
QE(df = 10) = 28.3251, p-val = 0.0016

Test of Moderators (coefficient(s) 2,3):
QM(df = 2) = 12.2043, p-val = 0.0022

Model Results:

         estimate       se     zval    pval     ci.lb    ci.ub
intrcpt   -3.5455  29.0959  -0.1219  0.9030  -60.5724  53.4814
ablat     -0.0280   0.0102  -2.7371  0.0062   -0.0481  -0.0080  **
year       0.0019   0.0147   0.1299  0.8966   -0.0269   0.0307

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The estimated amount of residual heterogeneity is equal to $\hat{\tau}^2 = 0.1108$, suggesting that $(0.3132 - 0.1108)/.3132 = 65\%$ of the total amount of heterogeneity can be accounted for by including the two moderators in the model. However, while we can reject $\mathrm{H}_0\colon \beta_1 = \beta_2 = 0$ based on the omnibus test ($Q_M = 12.20$, $df = 2$, $p < 0.01$), only absolute latitude appears to have a significant influence on the effectiveness of the vaccine (i.e., for $\mathrm{H}_0\colon \beta_1 = 0$, we find $z = -2.74$ with $p < 0.01$, while for $\mathrm{H}_0\colon \beta_2 = 0$, we find $z = 0.13$ with $p = 0.90$). The test for residual heterogeneity is significant ($Q_E = 28.33$, $df = 10$, $p < 0.01$), possibly indicating that other moderators not considered in the model are influencing the vaccine effectiveness.

The results indicate that a one degree increase in absolute latitude corresponds to a change of −0.03 (95% CI: −0.01 to −0.05) units in terms of the average log relative risk. To facilitate the interpretation of the latitude moderator, we can obtain predicted average relative risks for various absolute latitude values, holding the year constant, for example, at 1970 (one could also consider dropping the year moderator from the model altogether). For this, we use the `predict()` function, using the `newmods` argument to specify the values of the moderators and the `transf` argument to specify a function for transforming the results (here, we use the `exp()` function for exponentiation of the predicted log relative risks). By setting `addx = TRUE`, the results are printed together with the moderator values used to obtain the predicted values.

```
R> predict(res, newmods = cbind(seq(from = 10, to = 60, by = 10), 1970),
+    transf = exp, addx = TRUE)

    pred se  ci.lb  ci.ub  cr.lb  cr.ub X.intrcpt X.ablat X.year
1 0.9345 NA 0.5833 1.4973 0.4179 2.0899         1      10   1970
2 0.7062 NA 0.5149 0.9686 0.3421 1.4579         1      20   1970
3 0.5337 NA 0.4196 0.6789 0.2663 1.0697         1      30   1970
4 0.4033 NA 0.2956 0.5502 0.1958 0.8306         1      40   1970
5 0.3048 NA 0.1916 0.4848 0.1369 0.6787         1      50   1970
6 0.2303 NA 0.1209 0.4386 0.0921 0.5761         1      60   1970
```

The results show the predicted average relative risks (`pred`) and the bounds of the corresponding 95% confidence intervals (`ci.lb` and `ci.ub`). The standard errors of the predicted values (`se`) are only provided when not using the `transf` argument.[3]

The average relative risk is not significantly different from 1 at 10° absolute latitude (95% CI: 0.58 to 1.50), indicating an equal infection risk on average for vaccinated and non-vaccinated individuals close to the equator. However, we see increasingly larger effects as we move further away from the equator. At 40°, the average infection risk is more than halved (95% CI: 0.30 to 0.55) for vaccinated individuals. At 60°, the risk of an infection in vaccinated individuals is on average only about a quarter as large (95% CI: 0.12 to 0.44).

More generally, Figure 2, shows a plot of the relative risk as a function of absolute latitude. The observed relative risks are drawn proportional to the inverse of the corresponding standard errors. The predicted effects with corresponding confidence interval bounds are also shown. For reasons to be discussed later, four studies (i.e., studies 4, 7, 12, and 13) are labeled with their study numbers in the figure. The figure (except for the labeling of the four studies) was created with the following code.

```
R> preds <- predict(res, newmods = cbind(0:60, 1970), transf = exp)
R> wi <- 1/sqrt(dat$vi)
R> size <- 0.5 + 3 * (wi - min(wi))/(max(wi) - min(wi))
R> plot(dat$ablat, exp(dat$yi), pch = 19, cex = size,
```

[3]The values under `cr.lb` and `cr.ub` denote the bounds of an approximate 95% credibility interval. The interval estimates where 95% of the true outcomes would fall in the hypothetical population of studies. This interval is calculated under the assumption that the value of τ^2 is known (and not estimated). A method for calculating a credibility interval that accounts for the uncertainty in the estimate of τ^2 will be implemented in the future.

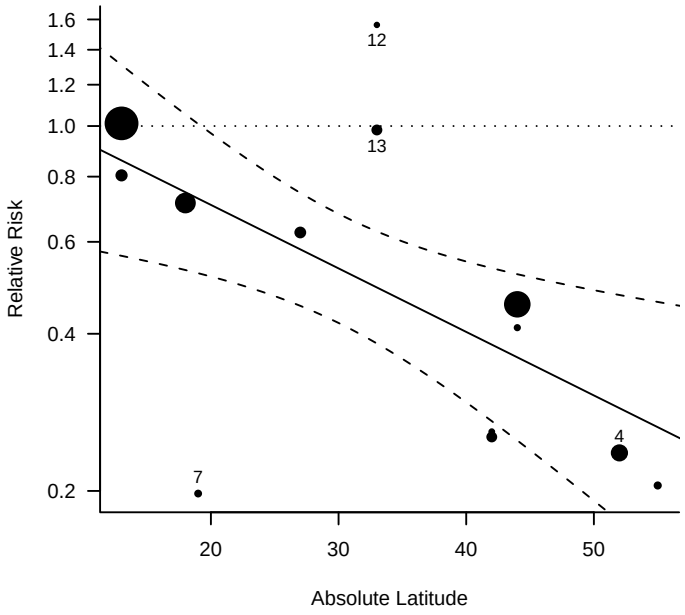


Figure 2: Relative risk of a tuberculosis infection versus absolute latitude of study location.

```
+    xlab = "Absolute Latitude", ylab = "Relative Risk",
+    las = 1, bty = "l", log = "y")
R> lines(0:60, preds$pred)
R> lines(0:60, preds$ci.lb, lty = "dashed")
R> lines(0:60, preds$ci.ub, lty = "dashed")
R> abline(h = 1, lty = "dotted")
```

Categorical moderator variables

Moderators are often categorical, either inherently or because the information provided in articles does not allow for more fine-grained coding. Therefore, subgrouping the studies based on the levels of a categorical moderator is a frequent practice in meta-analyses. One possibility is to fit a random-effects model separately within each level. For example, we can fit separate models within each level of the treatment allocation moderator with

```
R> rma(yi, vi, data = dat, subset = (alloc=="random"))
R> rma(yi, vi, data = dat, subset = (alloc=="alternate"))
R> rma(yi, vi, data = dat, subset = (alloc=="systematic"))
```

which illustrates the use of the `subset` argument (which can either be a logical vector as used here or a numeric vector indicating the indices of the observations to include). However, unless differences in the amount of heterogeneity are of interest or suspected to be present within the different levels, this is not an ideal approach (as τ^2 needs to be estimated separately

within each level based on an even smaller number of studies). Instead, we can fit a single mixed-effects model to these data with a dummy coded factor. First, we create the necessary dummy variables with

```
R> dat$a.random <- ifelse(dat$alloc == "random", 1, 0)
R> dat$a.alternate <- ifelse(dat$alloc == "alternate", 1, 0)
R> dat$a.systematic <- ifelse(dat$alloc == "systematic", 1, 0)
```

and then estimate separate effects for each factor level with

```
R> rma(yi, vi, mods = cbind(a.random, a.alternate, a.systematic),
+    intercept = FALSE, data = dat)
```

Instead of doing the coding manually, we can use the `factor()` function to handle the coding for us.

```
R> rma(yi, vi, mods = ~ factor(alloc) - 1, data = dat)
```

Either way, the following results will be obtained.

```
              estimate      se     zval    pval    ci.lb    ci.ub
a.random       -0.9658  0.2672  -3.6138  0.0003  -1.4896  -0.4420  ***
a.alternate    -0.5180  0.4412  -1.1740  0.2404  -1.3827   0.3468
a.systematic   -0.4289  0.3449  -1.2434  0.2137  -1.1050   0.2472
```

According to these results, only random treatment allocation leads to a significant treatment effect. However, to test whether the allocation factor is actually statistically significant, we need to use a different parameterization of the model by using

```
R> rma(yi, vi, mods = cbind(a.alternate, a.systematic), data = dat)
```

or, equivalently,

```
R> rma(yi, vi, mods = ~ factor(alloc), data = dat)
```

yielding the following results.[4]

```
Test of Moderators (coefficient(s) 2,3):
QM(df = 2) = 1.7675, p-val = 0.4132

              estimate      se     zval    pval    ci.lb    ci.ub
intrcpt        -0.9658  0.2672  -3.6138  0.0003  -1.4896  -0.4420  ***
a.alternate     0.4478  0.5158   0.8682  0.3853  -0.5632   1.4588
a.systematic    0.5369  0.4364   1.2303  0.2186  -0.3184   1.3921
```

[4]By default, the `factor()` function will use alternate instead of random treatment allocation as the reference level. The `relevel()` function can be used to set random allocation to the reference level (i.e., `rma(yi, vi, mods = ~ relevel(factor(alloc), ref = "random"), data = dat)`. However, regardless of which treatment allocation method is used as the reference level, the omnibus test of this factor will yield the same result.

Therefore, $\hat{\beta}_0 = -0.9658$ is the estimated average log relative risk for studies using random allocation, while $\hat{\beta}_1 = 0.4478$ and $\hat{\beta}_2 = 0.5369$ estimate how much larger the average log relative risks are when using alternate and systematic allocation, respectively (i.e., $\hat{\beta}_0 + \hat{\beta}_1 = -0.9658 + 0.4478 = -0.5180$, the estimated average log relative risk for studies using alternate allocation and $\hat{\beta}_0 + \hat{\beta}_2 = -0.9658 + 0.5369 = -0.4289$, the estimated average log relative risk for studies using systematic allocation). However, the test of $\mathrm{H}_0 : \beta_1 = \beta_2 = 0$ is not significant ($Q_M = 1.77$, $df = 2$, $p = 0.41$), suggesting that the type of allocation method does not actually influence the average effectiveness of the vaccine.

Knapp and Hartung adjustment

The next example illustrates the use of the Knapp and Hartung adjustment in the context of a model that includes both the allocation factor and the continuous absolute latitude moderator.

```
R> rma(yi, vi, mods = ~ factor(alloc) + ablat, data = dat, knha = TRUE)

Mixed-Effects Model (k = 13; tau^2 estimator: REML)

tau^2 (estimate of residual amount of heterogeneity): 0.1446 (SE = 0.1124)
tau (sqrt of the estimate of residual heterogeneity): 0.3803

Test for Residual Heterogeneity:
QE(df = 9) = 26.2034, p-val = 0.0019

Test of Moderators (coefficient(s) 2,3,4):
F(df1 = 3, df2 = 9) = 3.4471, p-val = = 0.0650

Model Results:

                         estimate      se     tval    pval    ci.lb    ci.ub
intrcpt                    0.2932  0.4188   0.7000  0.5016  -0.6543   1.2407
factor(alloc)random       -0.2675  0.3624  -0.7381  0.4793  -1.0873   0.5523
factor(alloc)systematic    0.0585  0.3925   0.1490  0.8849  -0.8294   0.9463
ablat                     -0.0273  0.0095  -2.8669  0.0186  -0.0488  -0.0058
```

The omnibus test ($\mathrm{H}_0 : \beta_1 = \beta_2 = \beta_3 = 0$) is now based on an F-distribution with $m = 3$ and $k - p = 9$ degrees of freedom, while a t-distribution with $k - p = 9$ degrees of freedom is now used as the reference distribution for tests and confidence intervals for the individual coefficients. Adding `btt = c(2, 3)` to the call would provide a test of $\mathrm{H}_0 : \beta_1 = \beta_2 = 0$, that is, an omnibus test only of the allocation factor (while controlling for the influence of absolute latitude).

Usually, the Knapp and Hartung adjustment will lead to more conservative p values, although this is not guaranteed in any particular case. In general though, the Type I error rate of the tests and the coverage probability of the confidence intervals will be closer to nominal when the adjustment is used. We can easily check this for the given data by repeatedly simulating a random moderator (which is unrelated to the outcomes by definition), recording the corresponding p value, and then calculating the empirical Type I error rate.

```
R> pval1 <- pval2 <- rep(NA, 10000)
R> for (i in 1:10000) {
+    xi <- rnorm(13)
+    pval1[i] <- rma(yi, vi, mods = xi, data = dat, method = "DL")$pval[2]
+    pval2[i] <- rma(yi, vi, mods = xi, data = dat, method = "DL",
+      knha = TRUE)$pval[2]
+  }
R> mean(pval1 < 0.05)
R> mean(pval2 < 0.05)
```

Note that the DerSimonian-Laird estimator is used, because it is non-iterative and therefore faster and guaranteed to provide an estimate for τ^2 (the REML estimator requires iterative estimation and can occasionally fail to converge; see Section 3.6 for some more technical details). The resulting empirical Type I error rates are approximately equal to 0.09 and 0.06, respectively. While the difference is not striking in this particular example, it illustrates how the Knapp and Hartung adjustment results in test statistics with closer to nominal properties.

3.5. Additional functions and methods

Table 1 provides an overview of the various functions and methods that can be used after fitting a model with the `rma()` function. Some of these additional functions will now be discussed.

Fitted/predicted values

The `fitted()` function can be used to obtain the fitted values for the k studies. The `predict()` function provides the fitted values in addition to standard errors and confidence interval bounds. As illustrated earlier, one can also use the `newmods` argument together with the `predict()` function to obtain predicted values for selected moderator values based on the fitted model. Note that for models without moderators, the fitted values are the same for all k studies (e.g., $\hat{\mu}$ in the random-effects model). The `predict()` function then only provides the fitted value once instead of repeating it k times.

For example, we can obtain an estimate of the average relative risk by first fitting a random-effects model to the log relative risks and then transforming the estimated average log relative risk (i.e., $\hat{\mu}$) back through exponentiation.

```
R> res <- rma(yi, vi, data = dat)
R> predict(res, transf = exp, digits = 2)

pred    se ci.lb ci.ub cr.lb cr.ub
0.49    NA  0.34  0.70  0.15  1.55
```

Raw and standardized residuals

Many meta-analyses will include at least a few studies yielding observed effects that appear to be outlying or extreme in the sense of being well separated from the rest of the data. Visual inspection of the data may be one way of identifying unusual cases, but this approach

Function	Description
`print()`	standard print method
`summary()`	alternative print method that also provides fit statistics
`coef()`	extracts the estimated model coefficients, corresponding standard errors, test statistics, p values, and confidence interval bounds
`vcov()`	extracts the variance-covariance matrix of the model coefficients
`fitstats()`	extracts the (restricted) log likelihood, deviance, AIC, and BIC
`fitted()`	fitted values
`predict()`	fitted/predicted values (with confidence intervals), also for new data
`blup()`	best linear unbiased predictions (BLUPs) of the true outcomes
`residuals()`	raw residuals
`rstandard()`	internally standardized residuals
`rstudent()`	externally standardized (studentized deleted) residuals
`hatvalues()`	extracts the diagonal elements of the hat matrix
`weights()`	extracts the weights used for model fitting
`influence()`	various case and deletion diagnostics
`leave1out()`	leave-one-out sensitivity analyses for fixed/random-effects models
`forest()`	forest plot
`funnel()`	funnel plot
`radial()`	radial (Galbraith) plot
`qqnorm()`	normal quantile-quantile plot
`plot()`	general plot function for model objects
`addpoly()`	function to add polygons to a forest plot
`ranktest()`	rank correlation test for funnel plot asymmetry
`regtest()`	regression tests for funnel plot asymmetry
`trimfill()`	trim and fill method
`confint()`	confidence interval for the amount of (residual) heterogeneity in random- and mixed-effects models (confidence intervals for the model coefficients can also be obtained)
`cumul()`	cumulative meta-analysis for fixed/random-effects models
`anova()`	model comparisons in terms of fit statistics and likelihoods
`permutest()`	permutation tests for model coefficients

Table 1: Functions and methods for fitted model objects created by the `rma.uni()` function.

may be problematic especially when dealing with models involving one or more moderators. Moreover, the studies included in a meta-analysis are typically of varying sizes (and hence, the sampling variances of the y_i values can differ considerably), further complicating the issue. A more formal approach is based on an examination of the residuals in relation to their corresponding standard errors.

Various types of residuals have been defined in the context of linear regression (e.g., Cook and Weisberg 1982), which can be easily adapted to the meta-analytic models. Most importantly, `rstandard()` and `rstudent()` provide internally and externally standardized residuals, respectively (`residuals()` provides the raw residuals). If a particular study fits the model, its standardized residual follows (asymptotically) a standard normal distribution. A large stan-

dardized residual for a study therefore may suggest that the study does not fit the assumed model (i.e., it may be an outlier).

For example, Figure 2 indicates that studies 7, 12, and 13 have observed outcomes that deviate noticeably from the model. However, the size of a residual must be judged relative to the precision of the predicted average effect for the corresponding study, while taking the amount of residual heterogeneity and the amount of sampling variability into consideration. Clearly, this is difficult to do by eye. On the other hand, the externally standardized residuals for this model can be easily obtained with

```
R> res <- rma(yi, vi, mods = cbind(ablat, year), data = dat)
R> rstudent(res)
```

```
     resid     se       z
1   0.2229 0.7486  0.2978
2  -0.2828 0.6573 -0.4303
3  -0.3826 0.7501 -0.5100
4  -1.0900 0.7768 -1.4032
.       .      .       .
7  -1.4061 0.5416 -2.5961
.       .      .       .
12  1.1864 0.8084  1.4677
13  0.7972 0.3742  2.1302
```

suggesting that only studies 7 and 13 have relatively 'large' residuals (even though the log relative risk for study 12 deviates considerably from the corresponding predicted average effect under the fitted model, the estimate for study 12 is also the least precise one of all 13 studies).

Influential case diagnostics

An outlying case may not be of much consequence if it exerts little influence on the results. However, if the exclusion of a study from the analysis leads to considerable changes in the fitted model, then the study may be considered to be influential. Case deletion diagnostics known from linear regression (e.g., Belsley *et al.* 1980; Cook and Weisberg 1982) can be adapted to the context of meta-analysis to identify such studies. The `influence()` function provides the following diagnostic measures for the various meta-analytic models:

- externally standardized residuals,
- DFFITS values,
- Cook's distances,
- covariance ratios,
- DFBETAS values,
- the estimates of τ^2 when each study is removed in turn,
- the test statistics for (residual) heterogeneity when each study is removed in turn,

- the diagonal elements of the hat matrix, and the
- the weights (in %) given to the observed outcomes during the model fitting.

For example, for the mixed-effects model with absolute latitude and publication year as moderators, we can obtain these diagnostic measures with

```
R> res <- rma(yi, vi, mods = cbind(ablat, year), data = dat)
R> inf <- influence(res)
R> inf
```

Instead of printing these results, we can use

```
R> plot(inf, plotdfb = TRUE)
```

to obtain two plots, the first with the various diagnostic measures except the DFBETAS values and the second with the DFBETAS values.

Figure 3 shows the first of these two plots, which suggests that studies 7 and 13 introduce some additional residual heterogeneity into the model (i.e., removing these studies in turn would yield considerably smaller estimates of τ^2), but only have a modest influence on the fit of the model (the plot of the Cook's distances shows this most clearly). On the other hand, removing study 4 would yield little change in the amount of residual heterogeneity, but its influence on the model fit is more considerable. Due to its large Cook's distance and hat value, it is also colored in red in the figure (a plot of the absolute latitudes against the publication years for the 13 studies (i.e., `plot(dat$ablat, dat$year)`) reveals the reason for the large influence of the 4th study).

For models without moderators, one can also use the `leave1out()` function to repeatedly fit the model, leaving out one study at a time. For example,

```
R> res <- rma(yi, vi, data = dat)
R> leave1out(res, transf = exp, digits = 3)
```

```
   estimate   se   zval  pval ci.lb ci.ub       Q    Qp  tau2     I2     H2
1     0.493   NA -3.722 0.000 0.340 0.716 151.583 0.000 0.336 93.226 14.762
2     0.520   NA -3.620 0.000 0.365 0.741 145.318 0.000 0.293 92.254 12.910
3     0.504   NA -3.692 0.000 0.350 0.725 150.197 0.000 0.321 92.935 14.155
4     0.533   NA -3.558 0.000 0.377 0.754  96.563 0.000 0.263 90.412 10.430
5     0.466   NA -3.984 0.000 0.320 0.678 151.320 0.000 0.328 92.763 13.819
6     0.491   NA -3.550 0.000 0.332 0.727 128.187 0.000 0.360 90.912 11.003
7     0.519   NA -3.631 0.000 0.365 0.740 145.830 0.000 0.293 92.278 12.950
8     0.452   NA -4.418 0.000 0.317 0.643  67.986 0.000 0.273 87.031  7.711
9     0.477   NA -3.769 0.000 0.324 0.701 152.205 0.000 0.349 93.213 14.735
10    0.520   NA -3.544 0.000 0.363 0.747 139.827 0.000 0.299 92.232 12.874
11    0.469   NA -3.871 0.000 0.319 0.688 151.466 0.000 0.340 91.811 12.211
12    0.468   NA -4.173 0.000 0.327 0.668 150.787 0.000 0.308 92.678 13.658
13    0.460   NA -4.191 0.000 0.319 0.661 149.788 0.000 0.304 92.344 13.062
```

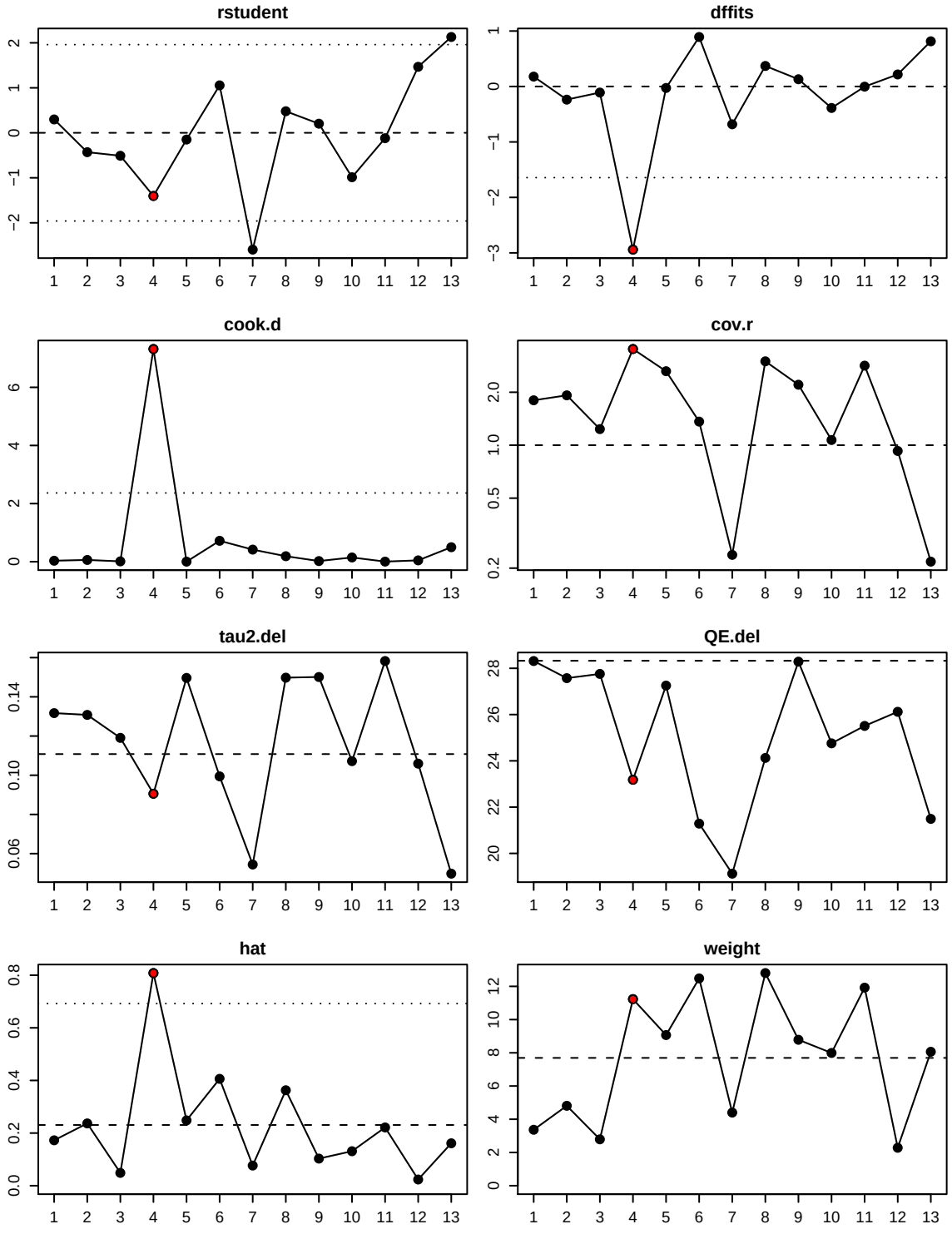


Figure 3: Plot of the externally standardized residuals, DFFITS values, Cook's distances, covariance ratios, estimates of τ^2 and test statistics for (residual) heterogeneity when each study is removed in turn, hat values, and weights for the 13 studies examining the effectiveness of the BCG vaccine for preventing tuberculosis.

shows the results from the random-effects model, leaving out one study at a time (the `transf` argument can again be used to specify a function which transforms the model estimate and the confidence interval bounds; the standard errors are then `NA`).

Plot functions (forest, funnel, radial, and Q-Q normal plots)

The **metafor** package provides several functions for creating plots that are frequently used in meta-analyses. Several examples are given in this section to illustrate how such plots can be created.

The use of the `forest()` function for creating forest plots from fitted model objects was already illustrated earlier (Figure 1). An additional example of a forest plot is shown in Figure 4 which shows how to create forest plots from individual effect size estimates and the corresponding sampling variances and illustrates the use of the `addpoly()` function for adding (additional) polygons to such plots. This is particularly useful to indicate the estimated effects for representative sets of moderator values or for subgroups of studies. The figure was created with the following code.

```
R> forest(dat$yi, dat$vi, atransf = exp, ylim = c(-3.5, 16),
+    at = log(c(0.05, 0.25, 1, 4, 20)), xlim = c(-9, 7),
+    slab = paste(dat$author, dat$year, sep = ", "))
R> res <- rma(yi, vi, mods = cbind(ablat), data = dat)
R> preds <- predict(res, newmods = c(10, 30, 50))
R> addpoly(preds$pred, sei = preds$se, atransf = exp,
+    mlab = c("10 Degrees", "30 Degrees", "50 Degrees"))
R> text(-9, 15, "Author(s) and Year", pos = 4, font = 2)
R> text(7, 15, "Relative Risk [95% CI]", pos = 2, font = 2)
R> abline(h = 0)
```

The `funnel()` function creates funnel plots (Light and Pillemer 1984; Sterne and Egger 2001), which can be useful for diagnosing the presence of heterogeneity and certain forms of publication bias (Rothstein *et al.* 2005). For models without moderators, the figure shows the observed outcomes on the horizontal axis against their corresponding standard errors (i.e., the square root of the sampling variances) on the vertical 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 $\pm 1.96 \cdot \text{SE}$, where SE is the standard error value from the vertical axis. For models involving moderators, the plot shows the residuals on the horizontal axis against their corresponding standard errors. A vertical line is drawn at zero with a pseudo confidence interval region given by $\pm 1.96 \cdot \text{SE}$.

Figure 5 shows two funnel plots, the first based on a random-effects model and the second based on a mixed-effects model with absolute latitude as moderator. These figures were created with the following code.

```
R> res <- rma(yi, vi, data = dat)
R> funnel(res, main = "Random-Effects Model")
R> res <- rma(yi, vi, mods = cbind(ablat), data = dat)
R> funnel(res, main = "Mixed-Effects Model")
```

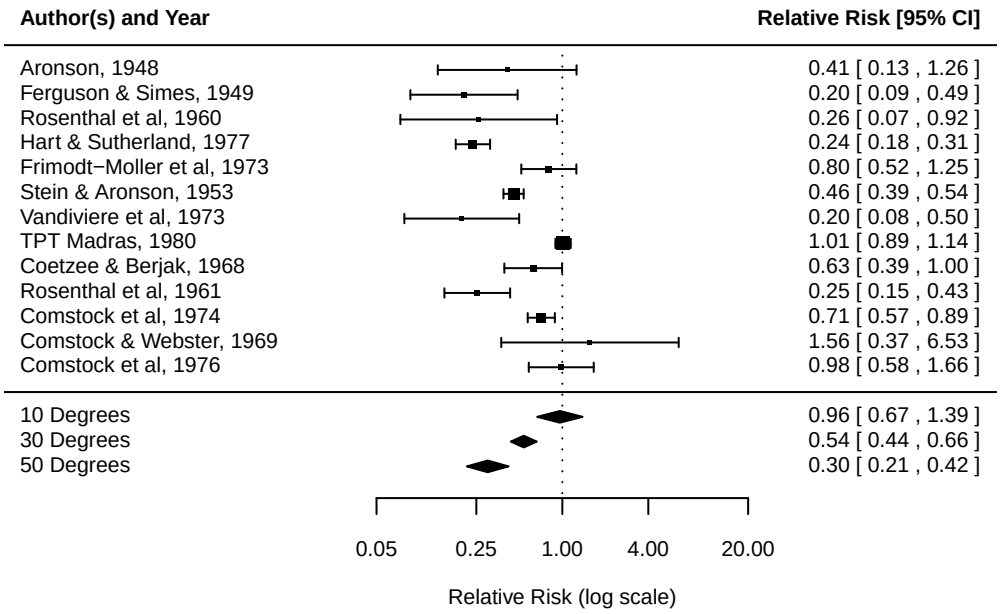


Figure 4: Forest plot showing the results of 13 studies examining the effectiveness of the BCG vaccine for preventing tuberculosis. The estimated average relative risk at 10, 30, and 50 degrees absolute latitude are indicated at the bottom of the figure.

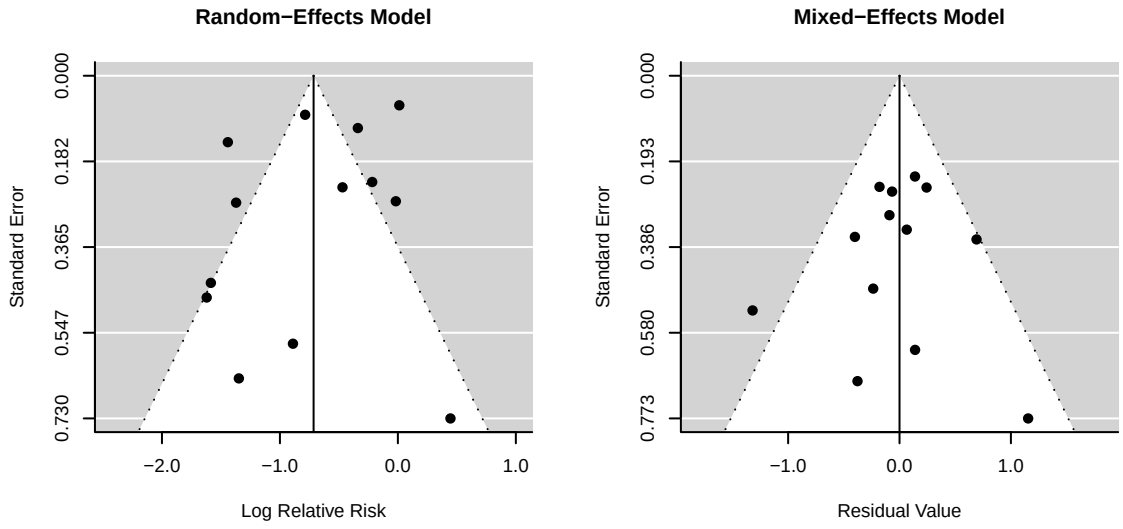


Figure 5: Funnel plot for a model without moderators (random-effects model) and a model with absolute latitude as moderator (mixed-effects model).

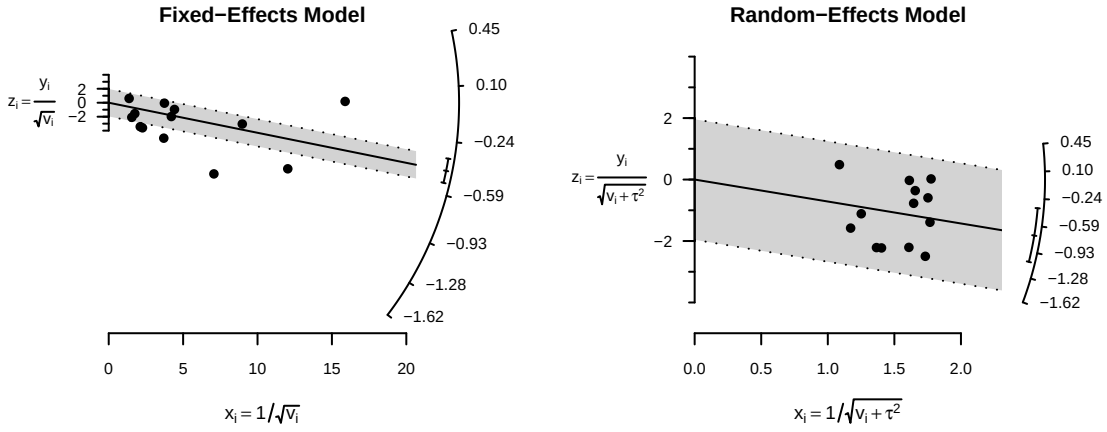


Figure 6: Radial plot for a fixed-effects and a random-effects model.

Radial (or Galbraith) plots were suggested by Rex Galbraith (1988a,b, 1994) as a way to assess the consistency of observed outcomes that have differing precisions (e.g., due to heteroscedastic sampling variances). For a fixed-effects model, the `radial()` function creates a plot showing the inverse of the standard errors on the horizontal axis (i.e., $1/\sqrt{v_i}$) against the individual observed outcomes standardized by their corresponding standard errors on the vertical axis (i.e., $y_i/\sqrt{v_i}$). On the right hand side of the plot, an arc is drawn. A line projected from $(0, 0)$ through a particular point within the plot onto this arc indicates the value of the observed outcome for that point. For a random-effects model, the function uses $1/\sqrt{v_i + \hat{\tau}^2}$ for the horizontal and $y_i/\sqrt{v_i + \hat{\tau}^2}$ for the vertical axis.

Figure 6 shows two examples of radial plots, one for a fixed- and the other for a random-effects model. The figures were created with the following code.

```
R> res <- rma(yi, vi, data = dat, method = "FE")
R> radial(res, main = "Fixed-Effects Model")
R> res <- rma(yi, vi, data = dat, method = "REML")
R> radial(res, main = "Random-Effects Model")
```

The `qqnorm()` function creates Q-Q normal plots which can be a useful diagnostic tool in meta-analyses (Wang and Bushman 1998). The plot shows the theoretical quantiles of a normal distribution on the horizontal axis against the observed quantiles of the (externally) standardized residuals on the vertical axis. For reference, a line is added to the plot with a slope of 1, going through the $(0, 0)$ point. By default, a pseudo confidence envelope is also added 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 and Weisberg 1982). The number of simulated sets can be controlled with the `reps` argument (`reps = 1000` by default). When `smooth = TRUE` (the default), the simulated bounds are smoothed with Friedman's Super-Smoother (see `help("supsmu")` for details). The `bass` argument can be set to a number

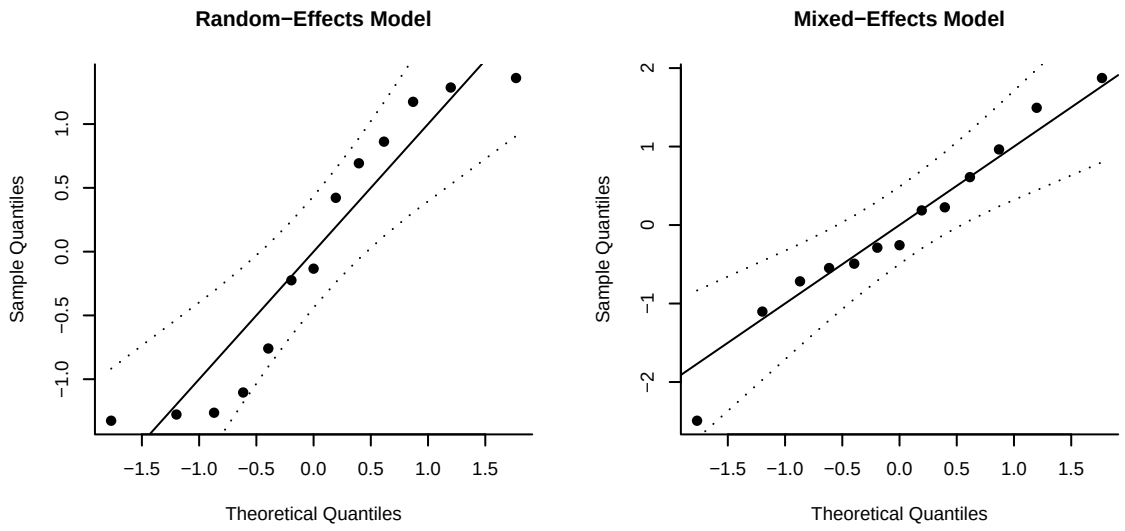


Figure 7: Q-Q normal plot for a model without moderators (random-effects model) and a model with absolute latitude as moderator (mixed-effects model).

between 0 and 10, with higher numbers indicating increasing smoothness (`bass = 0` by default).

Figure 7 shows two examples of Q-Q normal plots, the first for a random-effects model and the second for a mixed-effects model with absolute latitude as a moderator. The figures were created with the following code.

```
R> res <- rma(yi, vi, data = dat)
R> qqnorm(res, main = "Random-Effects Model")
R> res <- rma(yi, vi, mods = cbind(ablat), data = dat)
R> qqnorm(res, main = "Mixed-Effects Model")
```

Tests for funnel plot asymmetry

As mentioned earlier, funnel plots can be a useful graphical device for diagnosing certain forms of publication bias. In particular, if studies with small and/or non-significant findings remain unpublished (and therefore are less likely to be included in a meta-analysis), then this may result in an asymmetric funnel plot (Light and Pillemer 1984; Sterne and Egger 2001; Rothstein *et al.* 2005). One may be able to detect such asymmetry by testing whether the observed outcomes (or residuals from a model with moderators) are related to their corresponding sampling variances, standard errors, or more simply, sample sizes.

Various tests for funnel plot asymmetry of this form 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 *et al.* (2001), Sterne and Egger (2005), Harbord *et al.* (2006), Peters *et al.* (2006), Rücker *et al.* (2008), and Moreno *et al.* (2009). The various versions of the regression test differ in terms of the model (either a 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 suggested outcome measure to use for the test (e.g., for 2×2 table data, one has the choice between various outcome measures, as described earlier).

The `ranktest()` and `regtest()` functions can be used to carry out the rank correlation and the regression tests. For the regression test, the arguments of the function are

```
regtest(x, model = "rma", predictor = "sei", ni = NULL, ...)
```

where `x` is a fitted model object. One can choose the model used for the test via the `model` argument, with `model = "lm"` for weighted regression with a multiplicative dispersion term or `model = "rma"` for the standard meta-analytic model (the default). 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 mixed-effects model, one should first fit a model with, for example, `method = "REML"` and then use the `regtest()` function on the fitted model object.

The predictor is chosen via the `predictor` argument, with `predictor = "sei"` for the standard error (the default), `predictor = "vi"` for the sampling variance, `predictor = "ni"` for the sample size, and `predictor = "ninv"` for the inverse of the sample size. The fitted model object will automatically contain information about the sample sizes when `measure` was *not* equal to `"GEN"` during the initial model fitting. The sample sizes can also be supplied via the `ni` argument when `measure = "GEN"` during the initial model fitting.

For example, to carry out the regression test with a weighted regression model using the standard error as the predictor, we would use the following code.

```
R> res <- rma(yi, vi, data = dat)
R> regtest(res, model = "lm")

Regression Test for Funnel Plot Asymmetry

model:     weighted regression with multiplicative dispersion
predictor: standard error

t = -1.4013, df = 11, p = 0.1887
```

We may also want to control for the influence of potential moderators and use a meta-analytic mixed-effects model together with the sample size of the studies as a potential predictor. The regression test could then be carried out as follows.

```
R> res <- rma(ai = tpos, bi = tneg, ci = cpos, di = cneg, data = dat,
+    measure = "RR", mods = cbind(ablat, year))
R> regtest(res, predictor = "ni")

Regression Test for Funnel Plot Asymmetry
```

```
model:     mixed-effects meta-regression model
predictor: total sample size

z = 0.7470, p = 0.4550
```

Neither test suggests asymmetry in the respective funnel plots, although the results must be treated with some caution. The references given earlier provide more details regarding these tests.

Trim and fill method

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. The trim and fill method can only be used in the context of the 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).

After fitting either a fixed- or a random-effects model, one can use the `trimfill()` function to carry out the trim and fill method on the fitted model object. The syntax of the function is given by

```
trimfill(x, estimator = "L0", side = NULL, maxit = 50, verbose = FALSE, ...)
```

where `x` again denotes the fitted model object, `estimator` is used to choose between the `"L0"` or `"R0"` estimator for the number of missing studies (see references), `side` is an argument to indicate on which side of the funnel plot the missing studies should be imputed (if `side = NULL`, the side is chosen within the function depending on the results of the regression test), `maxit` denotes the maximum number of iterations to use for the trim and fill method, and `verbose` can be set to `TRUE` to obtain information about the evolution of the algorithm underlying the trim and fill method.

To illustrate the use of the function, we can fit a fixed-effects model to the BCG vaccine data and then use the trim and fill method to obtain the estimated number of missing studies. The model object is automatically augmented with the missing data and can then be printed.

```
R> res <- rma(yi, vi, data = dat, method = "FE")
R> rtf <- trimfill(res)
R> rtf

Estimated number of missing studies on the right side: 4

Fixed-Effects Model (k = 17)

Test for Heterogeneity:
Q(df = 16) = 262.7316, p-val < .0001
```

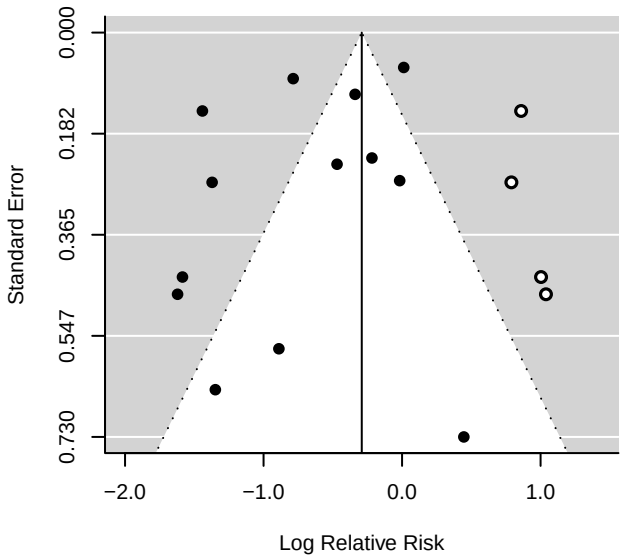


Figure 8: Funnel plot with filled-in data based on the trim and fill method.

```
Model Results:

estimate       se     zval     pval    ci.lb    ci.ub
 -0.2910   0.0383  -7.6057   <.0001  -0.3660  -0.2160      ***
```

Even though the estimated effect of the vaccine is smaller with the missing studies filled in, the results still indicate that the effect is statistically significant. A funnel plot with the filled-in studies can now be obtained with

```
R> funnel(rtf)
```

Figure 8 shows the resulting funnel plot with the filled-in data.

As a final note to the issue of funnel plot asymmetry and publication bias, it is worth mentioning the **copas** package (Carpenter and Schwarzer 2009), which can be used together with the **meta** package (Schwarzer 2010) and provides additional methods for modeling and adjusting for bias in a meta-analysis via selection models (Copas 1999; Copas and Shi 2000, 2001).

Cumulative meta-analysis

In a cumulative meta-analysis, an estimate of the average effect is obtained sequentially as studies are added to the analysis in (typically) chronological order (Chalmers and Lau 1993; Lau *et al.* 1995). Such analyses are usually conducted retrospectively (i.e., after all of the studies have already been conducted), but may be planned prospectively (Whitehead 1997). The method exemplifies how evidence regarding a particular effect evolves over time.

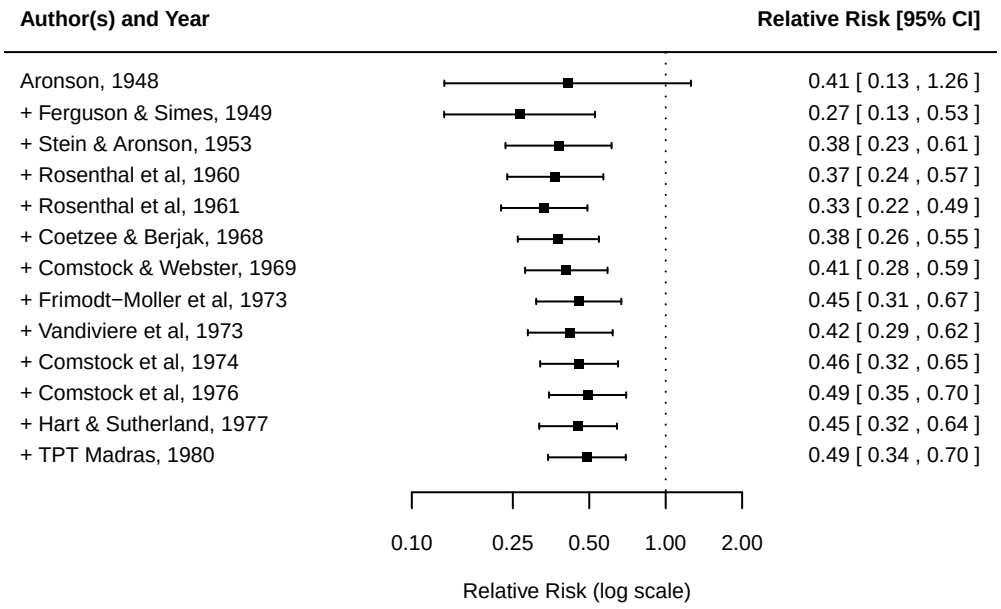


Figure 9: Forest plot, showing the results from a cumulative meta-analysis of 13 studies examining the effectiveness of the BCG vaccine for preventing tuberculosis.

Cumulative meta-analyses can also be carried out with the **metafor** package using the `cumul()` function. The function takes as its first argument a fitted model object (either a fixed- or a random-effects model) and then refits the same model k times adding one study at a time. The `order` argument of the function can be set equal to a vector with indices giving the desired order for the cumulative meta-analysis. The results can either be printed or passed on to the `forest()` function, which then creates a cumulative forest plot. For example,

```
R> res <- rma(yi, vi, data = dat, slab = paste(author, year, sep=", "))
R> rcu <- cumul(res, order = order(dat$year))
R> forest(rcu, xlim = c(-6, 3), atransf = exp,
+    at = log(c(0.10, 0.25, 0.5, 1, 2)))
R> text(-6, 15, "Author(s) and Year", pos = 4, font = 2)
R> text(3, 15, "Relative Risk [95% CI]", pos = 2, font = 2)
```

results in the forest plot shown in Figure 9. Note that the study labels must already be specified via the `rma()` function (via argument `slab`), so that they can be properly ordered by the `cumul()` function. Although the effectiveness of the vaccine appears to be decreasing over time, this finding is related to the fact that the more recent studies were conducted closer to the equator.

Model fit statistics

Model fit statistics can be obtained via the `fitstats()` function. In particular, the (restricted) log likelihood, deviance (-2 times the log likelihood), AIC, and BIC are provided.

The unrestricted log likelihood is computed (and used for calculating the deviance, AIC, and BIC), unless REML estimation is used to estimate τ^2 (in which case the restricted log likelihood is used). Note that τ^2 is counted as an additional parameter in the calculation of the AIC and BIC in random/mixed-effects models.

Likelihood ratio tests

As an alternative to Wald tests, one can conduct full versus reduced model comparisons via likelihood ratio tests with the `anova()` function. The function provides information about the fit statistics of two models and the corresponding results from a likelihood ratio test. Obviously, the two models must be based on the same set of data and should be nested for the likelihood ratio test to make sense. Also, likelihood ratio tests are not meaningful when using REML estimation and the two models differ with respect to their fixed effects. Therefore, to test moderator variables via likelihood ratio tests, one must switch to maximum likelihood (ML) estimation.

To illustrate the use of the `anova()` function, suppose that we would like to conduct a likelihood ratio test of the absolute latitude moderator.

```
R> res1 <- rma(yi, vi, mods = cbind(ablat), data = dat, method = "ML")
R> res2 <- rma(yi, vi, data = dat, method = "ML")
R> anova(res1, res2)

          df     AIC     BIC   logLik    LRT   pval       QE  tau^2    VAF
Full       3 21.3713 23.0662  -7.6857                30.7331 0.0344
Reduced    2 29.3302 30.4601 -12.6651 9.9588 0.0016 152.2330 0.2800 87.73%
```

The null hypothesis $H_0: \beta_1 = 0$ is rejected ($LRT = 9.96$, $df = 1$, $p < 0.002$), again suggesting that absolute latitude does have an influence on the average effectiveness of the vaccine. The function also provides the estimates of τ^2 from both models and indicates how much of the (residual) heterogeneity in the reduced model is accounted for in the full model (i.e., $100\% \times (\hat{\tau}^2_R - \hat{\tau}^2_F)/\hat{\tau}^2_R$, where $\hat{\tau}^2_F$ and $\hat{\tau}^2_R$ are the estimated values of τ^2 in the full and reduced model, respectively). In this example, approximately 88% of the total heterogeneity in the true effects is accounted for by the absolute latitude moderator.

In principle, one can also consider likelihood ratio tests for (residual) heterogeneity (i.e., for testing $H_0: \tau^2 = 0$) 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"` (while keeping the fixed effects the same in both models). The p value from the test would then be based on a χ^2 distribution with 1 degree of freedom, but actually needs to be adjusted for the fact that the parameter (i.e., τ^2) falls on the boundary of the parameter space under the null hypothesis. However, the Q-test usually keeps better control of the Type I error rate anyway and therefore should be preferred (see Viechtbauer 2007b for more details).

Permutation tests

Follmann and Proschan (1999) and Higgins and Thompson (2004) have suggested permutation tests of the model coefficients in the context of meta-analysis as an alternative approach to the standard (Wald and likelihood ratio) tests which assume normality of the observed effects (as

well as the true effects in random/mixed-effects models) and rely on the asymptotic behavior of the test statistics.

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 twice the proportion of times that the test statistic under the permuted data is as extreme or more extreme than under the actually observed data.

For models with moderators, the permutation test is carried out by permuting the rows of the design matrix. The (two-sided) p value for a particular model coefficient is then equal to twice 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 more extreme than the actually observed one.

Permutation tests can be carried out with the `permutest()` function. The function takes as its first argument a fitted model object. 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), especially with respect to the possible number of permutations of the design matrix. For example, for $k = 5$, there are only 120 possible permutations of the design matrix (32 possible permutations of the signs). For $k = 8$, there are already 40,320 (256). And for $k = 10$, there are 3,628,800 (1024). 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 (`iter = 1000` by default). Therefore, running the function twice on the same data will then yield (slightly) different p values. Setting `iter` sufficiently large ensures that the results become stable. For example,

```
R> res <- rma(yi, vi, data = dat)
R> permutest(res, exact = TRUE)

Running 8192 iterations for exact permutation test.

Model Results:

         estimate      se     zval   pval*    ci.lb    ci.ub
intrcpt   -0.7145  0.1798  -3.9744  0.0015  -1.0669  -0.3622  **

---
Signif. codes:  0 '**' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

indicates that the null hypothesis $\mathrm{H}_0 : \mu = 0$ can be rejected with $p = 0.0015$ after carrying out an exact permutation test for the random-effects model. Similarly, for a mixed-effects model,

```
R> res <- rma(yi, vi, mods = cbind(ablat, year), data = dat)
```

```
R> permres <- permutest(res, iter = 10000, retpermdist = TRUE)
R> permres

Running 10000 iterations for approximate permutation test.

Test of Moderators (coefficient(s) 2,3):
QM(df = 2) = 12.2043, p-val* = 0.0232

Model Results:

         estimate       se     zval   pval*     ci.lb    ci.ub
intrcpt   -3.5455  29.0959  -0.1219  0.9150  -60.5724  53.4814
ablat     -0.0280   0.0102  -2.7371  0.0224   -0.0481  -0.0080  *
year       0.0019   0.0147   0.1299  0.8950   -0.0269   0.0307

---
Signif. codes:  0 '**' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

shows the results for an approximate permutation test with 10000 iterations (an exact test is not feasible here, as it would require more than 6×10^9 model fits). Although the conclusions are unchanged for the random- and the mixed-effects models, we see that the permutation-based p values are larger (i.e., more conservative) when compared to the corresponding results presented earlier.

By setting `retpermdist = TRUE`, the permutation distributions of the test statistics for the individual coefficients and the omnibus test are returned together with the object. These elements are named `zval.perm` and `QM.perm` and can be used to examine the permutation distributions. For example, Figure 10 shows a histogram of the permutation distribution of the test statistic for absolute latitude, together with the standard normal density (in red) and a kernel density estimate of the permutation distribution (in blue). The figure shows that the tail area under the permutation distribution is larger than under the standard normal density (hence, the larger p value in this case). Figure 10 was created with the following code (leaving out the code for the annotations):

```
R> hist(permres$zval.perm[,2], breaks = 140, freq = FALSE, xlim = c(-5, 5),
+    ylim = c(0, 0.4), main = "", xlab = "Value of Test Statistic")
R> abline(v = res$zval[2], lwd = 2, lty = "dashed")
R> abline(v = 0, lwd = 2)
R> curve(dnorm, from = -5, to = 5, add = TRUE, lwd = 2,
+    col = rgb(1, 0, 0, alpha = 0.7))
R> lines(density(permres$zval.perm[,2]), lwd = 2,
+    col = rgb(0, 0, 1, alpha = 0.7))
```

Like the Knapp and Hartung adjustment, permutation tests lead to test statistics with better control of the Type I error rate. To examine this, a simulation was conducted like in Section 3.4, randomly generating values for an unrelated moderator and then testing its significance via a permutation test. This was repeated 10000 times with 5000 iterations for the

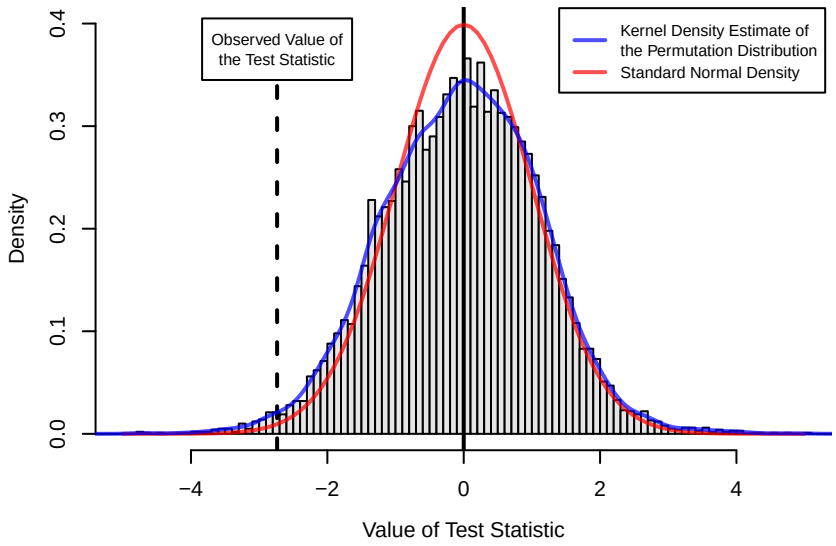


Figure 10: Permutation distribution of the test statistic for absolute latitude.

permutation test (for a total of 5×10^7 model fits). The empirical Type I error rate of the test was equal to 0.05, indicating nominal performance of the test for the BCG vaccine data.

Best linear unbiased predictions

For random/mixed-effects models, the `blup()` function calculates the best linear unbiased predictions (BLUPs) of the true outcomes by combining the fitted values (which are based only on the fixed effects of the model) and the estimated contributions of the random effects (e.g. Morris 1983; Raudenbush and Bryk 1985; Robinson 1991). Corresponding standard errors and prediction interval bounds are also provided.[5] There are various ways of interpreting these values. Essentially, the BLUPs are the predicted θ_i values in the random- and mixed-effects models (Equations 2 and 3) for the k studies included in the meta-analysis. From a Bayesian perspective, the BLUPs can also be regarded as the means (and due to normality also the modes) of the posterior distributions of the θ_i values under vague priors on the model coefficients.

What is most notable about the BLUPs is their "shrinkage" behavior, which is most easily illustrated in the context of a random-effects model. Suppose that $\tau^2 = 0$ and we want to obtain estimates of the study-specific θ_i values. Then the best estimates would all be equal to $\hat{\mu} \equiv \hat{\theta}$, since homogeneity implies that there is only one true effect. On the other hand, when τ^2 is very large, then $\hat{\mu}$ contains very little information about the location of the study-specific true effects. Instead, we should then place more emphasis on the observed estimates, especially for studies with small sampling variances (where the observed estimates will tend to be close to the corresponding θ_i values). In general, the BLUPs will fall somewhere in between y_i and $\hat{\mu}$, depending on the size of τ^2 and the amount of sampling variability. In that

[5]To be precise, it should be noted that the function actually calculates empirical BLUPs (eBLUPs), since the predicted values are a function of τ^2, which must be estimated based on the data. Following Kackar and Harville (1981), we know however that the eBLUPs are unbiased and approach the real BLUPs asymptotically.

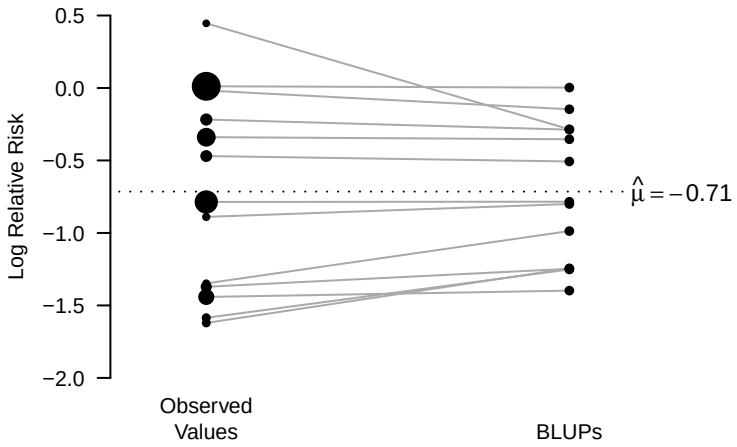


Figure 11: Plot showing the observed log relative risks and the corresponding best linear unbiased predictions (BLUPs) for the 13 studies examining the effectiveness of the BCG vaccine for preventing tuberculosis (with the observed values drawn proportional to the precision of the estimates).

sense, the observed data are "shrunken" towards $\hat{\mu}$.

To demonstrate this point, we can obtain the BLUPs based on a random-effects model with

```
R> res <- rma(yi, vi, data = dat)
R> blup(res)

      pred     se   pi.lb   pi.ub
1  -0.8002 0.4099 -1.6037  0.0032
2  -1.2517 0.3532 -1.9440 -0.5594
3  -0.9869 0.4348 -1.8392 -0.1346
4  -1.3979 0.1376 -1.6675 -1.1283
5  -0.2874 0.2113 -0.7015  0.1268
.        .      .       .       .
13 -0.1467 0.2434 -0.6239  0.3304
```

A plot of the observed log relative risks and the corresponding BLUPs is shown in Figure 11, clearly demonstrating the shrinkage behavior of the BLUPs, especially for the smaller studies (i.e., studies with larger sampling variances).

3.6. Technical details

This section concludes with some technical details about the algorithms and methods used in the **metafor** package, in particular with respect to the `rma()` function. Some issues related to the assumptions of the model underlying the `rma()` function are also touched on.

While the HS, HE, DL, and SJ estimators of τ^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 and 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 τ^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 case). 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)
```

when calling the `rma()` function. 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). Information on the evolution of the algorithm can be obtained with `control = list(verbose = TRUE)`.

All of the heterogeneity estimators except SJ can in principle yield negative estimates for the amount of (residual) heterogeneity. However, negative estimates of τ^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 τ^2 equal to some other value besides zero with `control = list(tau2.min = value)`.[6]

The Hunter-Schmidt estimator for the amount of heterogeneity is defined in Hunter and Schmidt (2004) 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 and is implemented in the `rma()` function.

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 τ^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 τ^2 turns out to be zero.

A "singular matrix" error when using the function indicates that there is a linear relationship between the moderator variables included in the model. For example, two moderators that correlated perfectly would cause this error. Deleting (redundant) moderator variables from the model as needed should solve this problem.

Finally, some words of caution about the assumptions underlying the models are warranted:

- The sampling variances (i.e., the v_i values) are treated as if they were known constants. Since this is usually only asymptotically true, this 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

[6]Technically, the random/mixed-effects models only require $v_i + \tau^2$ to be larger than zero to avoid marginal variances that are negative. Therefore, one could consider setting `tau2.min` just slightly larger than `-min(dat$vi)`. However, since τ^2 is usually of inherent interest for interpretational purposes, we generally prefer to avoid the possibility of a negative variance estimate.

the sampling variance estimates is small). Certain outcome measures (e.g., the arcsine transformed risk difference and Fisher's r-to-z transformed correlation coefficient) are based on variance stabilizing transformations that also help to make the assumption of known sampling variances more reasonable.

- When fitting a mixed/random-effects model, τ^2 is estimated and then treated as a known constant thereafter. This ignores the uncertainty in the estimate of τ^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 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) also 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.

These concerns apply in particular to the standard (i.e., Wald and likelihood ratio) tests (and the corresponding confidence intervals). Permutation tests may provide better control of the Type I error rate when some of these assumptions are violated, but more research is needed to determine the properties of such tests for different outcome measures.

4. Validation of the package

The functions in the **metafor** package have been validated to the extent possible (i.e., when corresponding analyses could be carried out) by comparing the results provided by the **metafor** package with those provided by other software packages for several data sets (including the BCG vaccine data set described in the present article).

In particular, results were compared with those provided by the `metan`, `metareg`, `metabias`, and `metatrim` commands in Stata (StataCorp. 2007; see Sterne 2009 for more details on these commands). Results were also compared with those provided by SAS (SAS Institute Inc. 2003) using the `proc mixed` command (van Houwelingen *et al.* 2002), by SPSS (SPSS Inc. 2006) using the macros described in Lipsey and Wilson (2001), and by the **meta** (Schwarzer 2010) and **rmeta** (Lumley 2009) packages in R (R Development Core Team 2010). Results either agreed completely or fell within a margin of error expected when using numerical methods.

5. Comparison between packages

Several packages for conducting meta-analyses are currently available for R via CRAN. These include the **metafor** (Viechtbauer 2010), **meta** (Schwarzer 2010), and **rmeta** (Lumley 2009) packages, all of which could be considered "general purpose" meta-analysis packages (i.e., they can be used for arbitrary effect size or outcome measures).[7] This section provides a brief

[7]Other packages include **catmap**, **metaMA**, **metacor**, **MADAM**, **MAMA**, **MAc**, **MAd**, and **psychometric**. These packages are restricted to special types of meta-analytic applications or particular outcome measures.

	metafor	**meta**	**rmeta**
Model fitting:			
Fixed-effects models	yes	yes	yes
Random-effects models	yes	yes	yes
Heterogeneity estimators	various	DL	DL
Mantel-Haenszel method	yes	yes	yes
Peto's method	yes	yes	no
Plotting:			
Forest plots	yes	yes	yes
Funnel plots	yes	yes	yes
Radial plots	yes	yes	no
L'Abbé plots	no	yes	no
Q-Q normal plots	yes	no	no
Moderator analyses:			
Categorical moderators	multiple	single[1]	no
Continuous moderators	multiple	no	no
Mixed-effects models	yes	no	no
Testing/Confidence Intervals:			
Knapp & Hartung adjustment	yes	no	no
Likelihood ratio tests	yes	no	no
Permutation tests	yes	no	no
Other:			
Leave-one-out analysis	yes	yes	no
Influence diagnostics	yes	no	no
Cumulative meta-analysis	yes	yes	yes
Tests for funnel plot asymmetry	yes	yes	no
Trim and fill method	yes	yes	no
Selection models	no	yes[2]	no

Table 2: Comparison of the capabilities of the **metafor**, **meta**, and **rmeta** packages for conducting meta-analyses in R. Notes: (1) Only fixed-effects with moderators model. (2) When used together with the **copas** (Carpenter and Schwarzer 2009) package.

comparison between these packages in terms of their current capabilities.

All three packages allow the user to fit fixed- and random-effects models (without moderators). The user can either specify the observed outcomes and the corresponding sampling variances (or standard errors) directly (via the `metagen()` function in **meta** and the `meta.summaries()` function in **rmeta**) or can provide the necessary information (e.g., 2×2 table data) so that the outcomes (e.g., log relative risks) and sampling variances are automatically computed inside of the functions (`metabin()`, `metacont()`, and `metaprop()` in **meta**; `meta.DSL()` and `meta.MH()`

in **rmeta**). For random-effects models, the **meta** and **rmeta** packages allow estimation of τ^2 only via the DerSimonian-Laird estimator, while the **metafor** package provides several estimator choices (see Section 3.3).

Forest and funnel plots can be created with all three packages. Radial plots are implemented in the **metafor** and **meta** packages. The **meta** package also provides L'Abbé plots (L'Abbé *et al.* 1987) (which will be added to the **metafor** package in the future). Q-Q normal plots can be obtained with the **metafor** package.

The **meta** package allows the user to specify a categorical moderator (via the `byvar` argument), which is then used for a moderator analysis based on a fixed-effects model. Mixed-effects models (involving a single or multiple categorical and/or continuous moderators) can only be fitted with the **metafor** package. Advanced methods for testing model coefficients and obtaining confidence intervals (i.e., the Knapp and Hartung adjustment and permutation tests) are also implemented only in this package.

A more detailed comparison of the capabilities of the packages can be found in Table 2.

6. Conclusions

The present article is meant to provide a general overview of the capabilities of the **metafor** package for conducting meta-analyses with R. The discussion was focused primarily on the `rma()` function, which allows for fitting fixed- and random/mixed-effects models with or without moderators via the usual mechanics of the general linear (mixed-effects) model.

Alternative methods for fitting the fixed-effects model for 2×2 table data are the Mantel-Haenszel and Peto's one-step method (Mantel and Haenszel 1959; Yusuf *et al.* 1985). The Mantel-Haenszel method is implemented in the `rma.mh()` function. It can be used to obtain an estimate of the overall odds ratio, relative risk, or risk difference. The method is particularly advantageous when aggregating a large number of tables with small sample sizes (the so-called sparse data or increasing strata case). When analyzing odds ratios, the Cochran-Mantel-Haenszel test (Mantel and Haenszel 1959; Cochran 1985) and Tarone's test for heterogeneity (Tarone 1985) are also provided.

Yet another method that can be used in the context of a meta-analysis of 2×2 tables is Peto's method (Yusuf *et al.* 1985), implemented in the `rma.peto()` function. The method provides a fixed-effects model estimate of the overall odds ratio. In sparse data situations and under certain conditions, the method has been shown to produce the least biased results with the most accurate confidence interval coverages (Bradburn *et al.* 2007), but can also be quite biased in other situations (Greenland and Salvan 1990).

It is important to note that all of these model fitting functions assume that the observed outcomes (or tables) are independent. At the very least, this implies that a particular participant should only contribute data once when calculating the observed outcomes. More complex models are necessary to deal with correlated outcomes and multivariate analyses. Functions to handle such situations are currently under development and will be included in the package at a later point.

Although the present article discusses only some of the functions and options of the **metafor** package, it should provide a starting point for those interested in exploring the capabilities of the package in more detail.

References

Bax L, Yu LM, Ikeda N, Tsuruta H, Moons KGM (2006). "Development and Validation of **MIX**: Comprehensive Free Software for Meta-Analysis of Causal Research Data." *BMC Medical Research Methodology*, **6**(50). URL http://www.biomedcentral.com/1471-2288/6/50/.

Begg CB, Mazumdar M (1994). "Operating Characteristics of a Rank Correlation Test for Publication Bias." *Biometrics*, **50**(4), 1088–1101.

Belsley DA, Kuh E, Welsch RE (1980). *Regression Diagnostics*. John Wiley & Sons, New York.

Berkey CS, Hoaglin DC, Mosteller F, Colditz GA (1995). "A Random-Effects Regression Model for Meta-Analysis." *Statistics in Medicine*, **14**(4), 395–411.

Borenstein M (2009). "Effect Sizes for Continuous Data." In H Cooper, LV Hedges, JC Valentine (eds.), *The Handbook of Research Synthesis and Meta-Analysis*, 2nd edition, pp. 221–235. Russell Sage Foundation, New York.

Borenstein M, Hedges L, Higgins J, Rothstein H (2005). ***Comprehensive Meta-Analysis***, *Version 2*. Biostat, Englewood, NJ. URL http://www.meta-analysis.com/.

Bradburn MJ, Deeks JJ, Berlin JA, Localio AR (2007). "Much Ado About Nothing: A Comparison of the Performance of Meta-Analytical Methods with Rare Events." *Statistics in Medicine*, **26**(1), 53–77.

Carpenter J, Schwarzer G (2009). "**copas**: Statistical Methods to Model and Adjust for Bias in Meta-Analysis." R package version 0.6-3, URL http://CRAN.R-project.org/package=copas.

Chalmers TC, Lau J (1993). "Meta-Analytic Stimulus for Changes in Clinical Trials." *Statistical Methods in Medical Research*, **2**(2), 161–172.

Cochran WG (1985). "Some Methods for Strengthening the Common χ^2 Tests." *Biometrics*, **10**(4), 417–451.

Colditz GA, Brewer TF, Berkey CS, Wilson ME, Burdick E, Fineberg HV, 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**(9), 698–702.

Cook RD, Weisberg S (1982). *Residuals and Influence in Regression*. Chapman and Hall, New York.

Copas J (1999). "What Works? Selectivity Models and Meta-Analysis." *Journal of the Royal Statistical Society*, **162**(1), 95–109.

Copas J, Shi JQ (2000). "Meta-Analysis, Funnel Plots and Sensitivity Analysis." *Biostatistics*, **1**(3), 247–262.

Copas J, Shi JQ (2001). "A Sensitivity Analysis for Publication Bias in Systematic Reviews." *Statistical Methods in Medical Research*, **10**(4), 251–265.

DerSimonian R, Laird N (1986). "Meta-Analysis in Clinical Trials." *Controlled Clinical Trials*, **7**(3), 177–188.

Duval SJ (2005). "The Trim and Fill Method." In HR Rothstein, AJ Sutton, M Borenstein (eds.), *Publication Bias in Meta-Analysis: Prevention, Assessment, and Adjustments*, pp. 127–144. John Wiley & Sons, Chichester, England.

Duval SJ, Tweedie RL (2000a). "A Nonparametric 'Trim and Fill' Method of Accounting for Publication Bias in Meta-Analysis." *Journal of the American Statistical Association*, **95**(449), 89–98.

Duval SJ, Tweedie RL (2000b). "Trim and Fill: A Simple Funnel-Plot-Based Method of Testing and Adjusting for Publication Bias in Meta-Analysis." *Biometrics*, **56**(2), 455–463.

Egger M, Smith GD, Schneider M, Minder C (1997). "Bias in Meta-Analysis Detected by a Simple, Graphical Test." *British Medical Journal*, **315**(7109), 629–634.

Fisher RA (1921). "On the 'Probable Error' of a Coefficient of Correlation Deduced From a Small Sample." *Metron*, **1**, 1–32.

Fleiss JL, Berlin JA (2009). "Effect Sizes for Dichotomous Data." In H Cooper, LV Hedges, JC Valentine (eds.), *The Handbook of Research Synthesis and Meta-Analysis*, 2nd edition, pp. 237–253. Russell Sage Foundation, New York.

Follmann DA, Proschan MA (1999). "Valid Inference in Random Effects Meta-Analysis." *Biometrics*, **55**(3), 732–737.

Freeman MF, Tukey JW (1950). "Transformations Related to the Angular and the Square Root." *Annals of Mathematical Statistics*, **21**(4), 607–611.

Galbraith RD (1988a). "Graphical Display of Estimates Having Differing Standard Errors." *Technometrics*, **30**(3), 271–281.

Galbraith RD (1988b). "A Note on Graphical Presentation of Estimated Odds Ratios from Several Clinical Trials." *Statistics in Medicine*, **7**(8), 889–894.

Galbraith RD (1994). "Some Applications of Radial Plots." *Journal of the American Statistical Association*, **89**(438), 1232–1242.

Ginsberg AM (1998). "The Tuberculosis Epidemic: Scientific Challenges and Opportunities." *Public Health Reports*, **113**(2), 128–136.

Glass GV (1976). "Primary, Secondary, and Meta-Analysis of Research." *Educational Researcher*, **5**(10), 3–8.

Greenland S, Salvan A (1990). "Bias in the One-Step Method for Pooling Study Results." *Statistics in Medicine*, **9**(3), 247–252.

Harbord RM, Egger M, Sterne JAC (2006). "A Modified Test for Small-Study Effects in Meta-Analyses of Controlled Trials with Binary Endpoints." *Statistics in Medicine*, **25**(20), 3443–3457.

Harville DA (1977). "Maximum Likelihood Approaches to Variance Component Estimation and to Related Problems." *Journal of the American Statistical Association*, **72**(358), 320–338.

Hedges LV (1989). "An Unbiased Correction for Sampling Error in Validity Generalization Studies." *Journal of Applied Psychology*, **74**(3), 469–477.

Hedges LV, Olkin I (1985). *Statistical Methods for Meta-Analysis*. Academic Press, San Diego, CA.

Hedges LV, Vevea JL (1998). "Fixed- and Random-Effects Models in Meta-Analysis." *Psychological Methods*, **3**(4), 486–504.

Higgins JPT, Thompson SG (2002). "Quantifying Heterogeneity in a Meta-Analysis." *Statistics in Medicine*, **21**(11), 1539–1558.

Higgins JPT, Thompson SG (2004). "Controlling the Risk of Spurious Findings from Meta-Regression." *Statistics in Medicine*, **23**(11), 1663–1682.

Hunter JE, Schmidt FL (2004). *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. 2nd edition. Sage, Newbury Park, CA.

Jennrich RI, Sampson PF (1976). "Newton-Raphson and Related Algorithms for Maximum Likelihood Variance Component Estimation." *Technometrics*, **18**(1), 11–17.

Kackar RN, Harville DA (1981). "Unbiasedness of Two-Stage Estimation and Prediction Procedures for Mixed Linear Models." *Communications in Statistics, Theory and Methods*, **10**(13), 1249–1261.

Knapp G, Hartung J (2003). "Improved Tests for a Random Effects Meta-Regression with a Single Covariate." *Statistics in Medicine*, **22**(17), 2693–2710.

Kontopantelis E, Reeves D (2009). "**MetaEasy**: A Meta-Analysis Add-In for Microsoft **Excel**." *Journal of Statistical Software*, **30**(7). URL http://www.jstatsoft.org/v30/i07/.

Krasopoulos G, Brister SJ, Beattie WS, Buchanan MR (2008). "Aspirin 'Resistance' and Risk of Cardiovascular Morbidity: Systematic Review and Meta-Analysis." *British Medical Journal*, **336**(7637), 195–198.

L'Abbé KA, Detsky AS, O'Rourke K (1987). "Meta-Analysis in Clinical Research." *Annals of Internal Medicine*, **107**(2), 224–233.

Laird NL, Mosteller F (1990). "Some Statistical Methods for Combining Experimental Results." *International Journal of Technology Assessment in Health Care*, **6**(1), 5–30.

Lau J, Schmid CH, Chalmers TC (1995). "Cumulative Meta-Analysis of Clinical Trials Builds Evidence for Exemplary Medical Care." *Journal of Clinical Epidemiology*, **48**(1), 45–57.

Lewis S, Clarke M (2001). "Forest Plots: Trying to See the Wood and the Trees." *British Medical Journal*, **322**(7300), 1479–1480.

Light RJ, Pillemer DB (1984). *Summing Up: The Science of Reviewing Research*. Harvard University Press, Cambridge, MA.

Lipsey MW, Wilson DB (eds.) (2001). *Practical Meta-Analysis*. Sage, Thousand Oaks, CA.

Lumley T (2009). "**rmeta**: Meta-Analysis." R package version 2.16, URL http://CRAN.R-project.org/package=rmeta.

Macaskill P, Walter SD, Irwig L (2001). "A Comparison of Methods to Detect Publication Bias in Meta-Analysis." *Statistics in Medicine*, **20**(4), 641–654.

Mantel N, Haenszel W (1959). "Statistical Aspects of the Analysis of Data from Retrospective Studies of Disease." *Journal of the National Cancer Institute*, **22**(4), 719–748.

Moreno SG, Sutton AJ, Ades AE, Stanley TD, Abrams KR, Peters JL, Cooper NJ (2009). "Assessment of Regression-Based Methods to Adjust for Publication Bias Through a Comprehensive Simulation Study." *BMC Medical Research Methodology*, **9**(2). URL http://www.biomedcentral.com/1471-2288/9/2.

Morris CN (1983). "Parametric Empirical Bayes Inference: Theory and Applications." *Journal of the American Statistical Association*, **78**(381), 47–55.

Olkin I (1995). "Meta-Analysis: Reconciling the Results of Independent Studies." *Statistics in Medicine*, **14**(5-7), 457–472.

Olkin I, Pratt JW (1958). "Unbiased Estimation of Certain Correlation Coefficients." *Annals of Mathematical Statistics*, **29**(1), 201–211.

Peters JL, Sutton AJ, Jones DR, Abrams KR, Rushton L (2006). "Comparison of Two Methods to Detect Publication Bias in Meta-Analysis." *Journal of the American Medical Association*, **295**(6), 676–680.

Petrin Z, Englund G, Malmqvist B (2008). "Contrasting Effects of Anthropogenic and Natural Acidity in Streams: A Meta-Analysis." *Proceedings of the Royal Society B: Biological Sciences*, **275**(1639), 1143–1148.

Raudenbush SW (2009). "Analyzing Effect Sizes: Random Effects Models." In H Cooper, LV Hedges, JC Valentine (eds.), *The Handbook of Research Synthesis and Meta-Analysis*, 2nd edition, pp. 295–315. Russell Sage Foundation, New York.

Raudenbush SW, Bryk AS (1985). "Empirical Bayes Meta-Analysis." *Journal of Educational Statistics*, **10**(2), 75–98.

R Development Core Team (2010). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. URL http://www.R-project.org/.

Roberts BW, Walton KE, Viechtbauer W (2006). "Patterns of Mean-Level Change in Personality Traits Across the Life Course: A Meta-Analysis of Longitudinal Studies." *Psychological Bulletin*, **132**(1), 1–25.

Robinson GK (1991). "That BLUP is a Good Thing: The Estimation of Random Effects." *Statistical Science*, **6**(1), 15–32.

Rosenberg MS, Adams DC, Gurevitch J (2000). ***MetaWin**: Statistical Software for Meta-Analysis Version 2*. Sinauer Associates, Sunderland, MA. URL http://www.metawinsoft.com/.

Rothstein HR, Sutton AJ, Borenstein M (eds.) (2005). *Publication Bias in Meta-Analysis: Prevention, Assessment, and Adjustments*. John Wiley & Sons, Chichester, England.

Rücker G, Schwarzer G, Carpenter J (2008). "Arcsine Test for Publication Bias in Meta-Analyses with Binary Outcomes." *Statistics in Medicine*, **27**(5), 746–763.

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**(5), 721–738.

SAS Institute Inc (2003). *SAS/STAT Software, Version 9.1*. SAS Institute Inc., Carry, NC. URL http://www.sas.com/.

Schwarzer G (2010). "**meta**: Meta-Analysis with R." R package version 1.6-0, URL http://CRAN.R-project.org/package=meta.

Shoemaker PJ, Tankard JW, Lasorsa DL (2003). *How to Build Social Science Theories*. Sage, Thousand Oaks, CA.

Sidik K, Jonkman JN (2005a). "A Note on Variance Estimation in Random Effects Meta-Regression." *Journal of Biopharmaceutical Statistics*, **15**(5), 823–838.

Sidik K, Jonkman JN (2005b). "Simple Heterogeneity Variance Estimation for Meta-Analysis." *Journal of the Royal Statistical Society C*, **54**(2), 367–384.

SPSS Inc (2006). *SPSS for Windows, Release 15*. SPSS Inc., Chicago, IL. URL http://www.spss.com/.

StataCorp (2007). *Stata Statistical Software: Release 9.2*. StataCorp LP, College Station, TX. URL http://www.stata.com/.

Sterne JAC (ed.) (2009). *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*. Stata Press, College Station, TX.

Sterne JAC, Egger M (2001). "Funnel Plots for Detecting Bias in Meta-Analysis: Guidelines on Choice of Axis." *Journal of Clinical Epidemiology*, **54**(10), 1046–1055.

Sterne JAC, Egger M (2005). "Regression Methods to Detect Publication and Other Bias in Meta-Analysis." In HR Rothstein, AJ Sutton, M Borenstein (eds.), *Publication Bias in Meta-Analysis: Prevention, Assessment, and Adjustments*, pp. 99–110. John Wiley & Sons, Chichester, England.

Tarone RE (1985). "On Heterogeneity Tests Based on Efficient Scores." *Biometrics*, **72**(1), 91–95.

The Cochrane Collaboration (2008). *Review Manager (**RevMan**) for Windows: Version 5.0*. The Nordic Cochrane Centre, Copenhagen, Denmark. URL http://www.cc-ims.net/revman/.

van Houwelingen HC, Arends LR, Stijnen T (2002). "Advanced Methods in Meta-Analysis: Multivariate Approach and Meta-Regression." *Statistics in Medicine*, **21**(4), 589–624.

Viechtbauer W (2005). "Bias and Efficiency of Meta-Analytic Variance Estimators in the Random-Effects Model." *Journal of Educational and Behavioral Statistics*, **30**(3), 261–293.

Viechtbauer W (2006). `mima`*: An S-PLUS/R Function to Fit Meta-Analytic Mixed-, Random-, and Fixed-Effects Models*. URL http://www.wvbauer.com/.

Viechtbauer W (2007a). "Confidence Intervals for the Amount of Heterogeneity in Meta-Analysis." *Statistics in Medicine*, **26**(1), 37–52.

Viechtbauer W (2007b). "Hypothesis Tests for Population Heterogeneity in Meta-Analysis." *British Journal of Mathematical and Statistical Psychology*, **60**(1), 29–60.

Viechtbauer W (2010). "**metafor**: Meta-Analysis Package for R." R package version 1.4-0, URL http://CRAN.R-project.org/package=metafor.

Wang MC, Bushman BJ (1998). "Using the Normal Quantile Plot to Explore Meta-Analytic Data Sets." *Psychological Methods*, **3**(1), 46–54.

Whitehead A (1997). "A Prospectively Planned Cumulative Meta-Analysis Applied to a Series of Concurrent Clinical Trials." *Statistics in Medicine*, **16**(24), 2901–2913.

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**(5), 335–371.

Zeileis A, Croissant Y (2010). "Extended Model Formulas in R: Multiple Parts and Multiple Responses." *Journal of Statistical Software*, **34**(1), 1–13. URL http://www.jstatsoft.org/v34/i01/.

Affiliation:

Wolfgang Viechtbauer
Department of Methodology and Statistics
School for Public Health and Primary Care
Maastricht University
P.O. Box 616
6200 MD Maastricht, The Netherlands
E-mail: wvb@wvbauer.com
URL: http://www.wvbauer.com/

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 36, Issue 3 *Submitted:* 2009-09-16
August 2010 *Accepted:* 2010-06-24