Journal of Educational and Behavioral Statistics
2017, Vol. 42, No. 2, pp. 206–242
DOI: 10.3102/1076998616674315

A Review of Meta-Analysis Packages in R

Joshua R. Polanin
Development Services Group, Inc.

Emily A. Hennessy
Emily E. Tanner-Smith
Vanderbilt University

Meta-analysis is a statistical technique that allows an analyst to synthesize effect sizes from multiple primary studies. To estimate meta-analysis models, the open-source statistical environment R is quickly becoming a popular choice. The meta-analytic community has contributed to this growth by developing numerous packages specific to meta-analysis. The purpose of this study is to locate all publicly available meta-analytic R packages. We located 63 packages via a comprehensive online search. To help elucidate these functionalities to the field, we describe each of the packages, recommend applications for researchers interested in using R for meta-analyses, provide a brief tutorial of two meta-analysis packages, and make suggestions for future meta-analytic R package creators.

Keywords: *meta-analysis; effect size; statistical software; R*

1. Introduction

Meta-analysis is a statistical technique that allows an analyst to combine effect sizes across multiple studies into one meaningful estimate. Compared to results from a single primary study, results from a meta-analysis provide greater generalizability, increased precision, and the ability to explore heterogeneity across studies (Borenstein, Hedges, Higgins, & Rothstein, 2010; Pigott, 2012). Meta-analysis has proven useful for policy makers and practitioners because the findings offer answers to ambiguous questions and synthesize large bodies of literature. As a result, published meta-analyses have increased exponentially over the past three decades (Williams, 2012).

Traditional meta-analysis where one wishes to pool effect size estimates using a simple inverse-variance-weighted mean (Hedges & Olkin, 1985) does not require sophisticated software or advanced computing power. An analyst may calculate a weighted average with relative ease by hand or through

ubiquitous spreadsheet software. Beyond the simplest meta-analytic tasks, however, it is more efficient and effective to use statistical software and preprogrammed applications. This is especially relevant for iterative or simulation-based techniques where closed-form calculations are not tenable or may introduce bias.

A fast-growing market for such software is the R statistical environment. R is a programming language and open-source software managed and maintained through the Comprehensive R network (R Core Team, 2015). R has become a popular and defining software choice for applied researchers and data scientists across many fields (Rexer Analytics, 2013). Meta-analysis packages built in R and using the R programming language follow this trend. Chen and Peace's (2013) and Schwarzer, Carpenter, and Rücker's (2015b) textbooks described many of the basic R meta-analytic packages, and Neupane, Richer, Bonner, Kibret, and Beyene (2014) detailed the R packages available to conduct a network meta-analysis. These previous works, however, have not synthesized the totality of meta-analytic R packages. Thus, the purpose of this article is to elucidate and describe R's meta-analytic packages available to researchers, highlight their capabilities, illustrate two popular meta-analysis packages' functionality, and provide recommendations for future improvement in packages.

2. Background

The traditional meta-analytic model estimates an average effect size by weighting each effect using the inverse of an effect size's variance (i.e., the squared standard error). In doing so, studies with a smaller variance, and all else equal a larger sample size, have a greater influence on the average effect size relative to studies with a larger variance. This is referred to as a fixed-effect estimate (Borenstein et al., 2010). An alternative and popular approach is to estimate a random-effects meta-analytic model, which, in addition to weighting each effect size by its inverse variance, takes into account the variance between studies. This can be represented as:

$$\theta = \frac{\sum W_i^* \times Y_i}{\sum W_i^*},$$

where θ represents the meta-analytic average effect size, Y_i is the effect size from study i, and W_i^* is the random-effects weight from study i. Under a random-effects model, each study's weight is represented as:

$$W_i^* = \frac{1}{V_i + \tau^2},$$

where V_i is the variance of the effect (squared standard error) from study i and τ^2 is the square of the between-studies variance estimate. Compared to a

fixed-effect meta-analytic estimate, the random-effects meta-analytic estimate will have a larger confidence interval and the weights of each study will be similar. It is possible to estimate τ using a closed-form solution (DerSimonian & Laird, 1986), but an iterative maximum likelihood estimator is less biased (Veroniki et al., 2016). Popular meta-analytic R packages such as `metafor` (Viechtbauer, 2015) or `metaSEM` (Cheung, 2015) will calculate the variance component using either estimator.

Recent advances in meta-analytic methods have moved beyond the traditional meta-analytic model. For instance, it is common for a single study to report multiple effect sizes within the same outcome domain. Under traditional meta-analytic assumptions, the analyst must represent each study with only one effect size, and, as such, analysts use *a priori* decision rules to choose one effect size or simply average all the effect sizes in a single study. Alternatively, one may use robust variance estimation to incorporate all effect sizes into one model (Hedges, Tipton, & Johnson, 2010). This method incorporates an estimation of the within-study covariance among the effect sizes to account for dependency. Although multivariate meta-analytic approaches have been suggested previously (e.g., Gasparrini, 2015), the robust variance model does not require that the user know the covariance structure among the effect sizes within each study. Tanner-Smith, Tipton, and Polanin (2016) suggested two approaches to weighting effect sizes, depending on the nature of the effect size dependencies, and the `robumeta` (Fisher & Tipton, 2015) package will estimate the robust variance meta-analytic model.

Similar advances have been made for a variety of common application issues. To conduct structural equation modeling using meta-analytic correlation matrices, Cheung (2015) has suggested a two-step modeling approach. The first step synthesizes complete, or approximately complete, correlation matrices. The second step then uses the synthesized correlation matrix to conduct the usual structural equation models such as factor analysis or latent variable modeling. Wilson, Polanin, and Lipsey (2016) extended this framework to accommodate complex data sets with multiple levels of dependency, and Cheung's (2015) `metaSEM` package will conduct a meta-analytic structural equation model. Another popular advance is the use of network model via network meta-analysis, where the goal is to contrast multiple interventions simultaneously. A simple network model can be represented using two studies. In Study 1, the authors contrast an intervention (A) to a no-intervention comparison group (B). In Study 2, the authors contrast the intervention in Study 1 (A) to another intervention (C). Using a network analysis, an analyst may then compare across all three contrasts (A, B, and C), despite the fact that these groups were not directly compared in primary studies. Although not as popular in the social or educational sciences, network meta-analyses are now common in medicine (Neupane, Richer, Bonner, Kibret, & Beyene, 2014).

Many other meta-analytic advances have been made in recent years. What follows is a presentation of many of these advances through the discussion of meta-analytic R packages. We first present a brief overview of how we found the packages, determined their eligibility, and then coded their capabilities. We then provide an overview of the packages as a whole. To make the packages easier to comprehend, we group the packages into categories based on their capabilities and discuss the packages among the categories and provide recommendations for future programmers. We end by presenting a brief tutorial on two popular meta-analytic packages: `metafor` (Viechtbauer, 2015) and `robumeta` (Fisher & Tipton, 2015).

3. Method

To locate and retrieve all R packages with meta-analytic functionality, we conducted a systematic and comprehensive search. The process began by searching three websites that warehouse R packages, the Comprehensive R Archive Network (CRAN) website (R Core Team, 2015), Revolution Analytics (2016), and Crantastic (Wickham & Maeland, 2016) using the search terms "meta-analysis," "meta analysis," or "systematic review." We also searched GitHub (GitHub, Inc., 2016) and Google Scholar (scholar.google.com). Expert meta-analysts as well as R programmers were also consulted. At the conclusion of the search, one author screened each R package for inclusion. The package was included if it provided any meta-analytic capability, regardless of its overall focus, capability, or limitations. For example, the package `CAMAN` (Schlattmann, Hoehne, & Verba, 2015) is primarily a suite of functions designed to conduct finite mixture models. The package also includes functionality to conduct basic meta-analyses of *p* values and therefore was included.

The next step was to code and extract characteristics from the packages. The goal of this process was to summarize the functionality available across the totality of meta-analytic packages as well as to determine any gaps in the applications, so that future programmers could ameliorate these issues. We also sought to provide context to the packages for novice users because it is often overwhelming to determine which packages provide certain functionality. As such, we coded a wide range of characteristics that may not be applicable to all packages, and instead illustrated the flexibility and diversity among the packages. The information we collected included capability to calculate effect sizes or conduct power analysis, whether and, if so, how the package addresses missing data, estimation of a fixed- or random-effects model, whether the package can accommodate dependent effect sizes, whether the package can estimate moderator analyses or sensitivity analyses, whether the package can assess publication bias, and any additional plotting capabilities. We also determined the name of the package's main function to help new users quickly leverage the package's capabilities. One author conducted all of the coding, but regular

coding meetings were held with the other members of the research team to discuss the nuances in the ways the package authors reported functionality.

After coding concluded, we conducted several analyses to describe the packages. First, we descriptively analyzed the sample, summarizing the main functionalities across the packages. Table 1 lists every meta-analysis package available as of September 2015. We also grouped the packages into nine primary categories according to their functions and capabilities: (1) general meta-analysis packages, (2) genome, (3) multivariate, (4) diagnostic, (5) specific, (6) network, (7) assessment of bias, (8) Bayesian, and (9) graphical user interface (GUI). Seven of the nine groups reflect the packages' main analytical functionality or characteristic. For instance, the general meta-analysis packages clearly intend to perform basic meta-analysis functions and many intend to house all the functionality a traditional meta-analyst needs. The other two, "genome" and "specific," represent two distinct package groups. The genome package group includes only packages that conduct genomic meta-analysis functions. This is a specific field of meta-analysis and therefore these packages should not be used for general purpose meta-analyses in the social or medical sciences. The specific package group constitutes packages that target very specialized meta-analysis procedures. Although the groupings reflect subjective categorizations, they provide a useful heuristic and are a tool to simplify the presentation and description of results. Finally, to provide usage context, we used the R package `cranlogs` (Csardi, 2015) to calculate each packages' average daily downloads by month.

The initial search for R packages concluded on January 30, 2015. The search was later updated, and the list is current through July 2, 2015. The coding process ended on September 29, 2015. We coded all functions into an Excel spreadsheet, and the summaries and graphs were created in R using the `ggplot2` package (Wickham, 2009). We coded the version number specific to the R package used at the time of coding although additional program updates may be available after coding ended. Unless otherwise stated, all packages were downloaded and implemented using R Version 3.1.2.

The search procedure yielded 106 total packages. We eliminated 38 packages from this list due to (a) duplication ($n = 32$) and (b) the package no longer being available for download (e.g., `RcmdrPlugin.Mac`; $n = 4$) or for not meeting our criteria ($n = 2$). Five additional packages were added through discussions with expert colleagues and after presenting initial findings at an R conference (Polanin & Hennessy, 2015). The total number of packages included in this analysis, therefore, was 63.

The highest number of packages created in a single year was in 2012 ($n = 12$), and the earliest package was created in 1999 (Figure 1). As outlined in Table 1, the reviewed packages share many similar functionalities. Most packages can calculate effect sizes (66.7%), run fixed-effect model (60.3%), or random-effects model (74.6%). A majority of the packages can create meta-analytic plots (74.6%), but many of these packages depend on a handful of other packages to

TABLE 1.

Package Characteristics

Package	Version	Title	Effect Size	Power	Missing Data	Dependent Effects	Fixed Effect	Random Effects	Moderator Analyses	Publication Bias	Sensitivity Analysis	Creates Plots	Primary Function
General meta-analysis													
CAMAN	0.7	Finite mixture models and meta-analysis tools						√	√				bivariate
epiR	0.9-62	Tools for the analysis of epidemiological data	√		√		√	√					epi.dsl
gmeta	2.2-3	Meta-analysis via a unified framework under confidence distribution	√			√	√	√				√	gmeta
Mac	1.1	Meta-analysis with correlations	√		√	√	√	√	√	√	√	√	mareg
Mad	0.8-2	Meta-analysis with mean differences	√		√	√	√	√	√	√		√	mareg
Meta	4.2-0	General package for meta-analysis	√		√		√	√	√	√	√	√	metacont
metacor	1.0-2	Meta-analysis of correlation coefficients	√				√	√					metacor.DSL
metafor	1.9-5	Meta-analysis package for R	√		√	√	√	√	√	√	√	√	rma
metaplus	0.7-1	Robust meta-analysis and meta-regression	√			√		√	√		√	√	metaplus
psychometric	2.2	Applied psychometric theory					√		√	√		√	MetaTable
rmeta	2.16	Meta-analysis	√				√	√		√	√	√	meta.MH

(continued)

TABLE 1. *(continued)*

Package	Version	Title	Effect Size	Power	Missing Data	Dependent Effects	Fixed Effect	Random Effects	Moderator Analyses	Publication Bias	Sensitivity Analysis	Creates Plots	Primary Function
Genome meta-analysis													
EasyStrata	8.6	Evaluation of stratified genome-wide association meta-analysis results				√	√	√				√	EasyStrata
Gap	1.1-16	Genetic analysis package	√			√	√	√				√	metap
GeneMeta	1.40.0	Meta-analysis for high throughput experiments	√				√	√			√	√	zScores
MADAM	1.2.2	This package provides some basic methods for meta-analysis										√	fisher.method
MAMA	2.2.1	Meta-analysis of MicroArray	√				√	√				√	ES.GeneMeta
metaARRAY	1.28.0	Integration of microarray data for meta-analysis						√					poe.em
MetABEL	0.2-0	Meta-analysis of genome-wide SNP association results	√									√	metagwa.files
MetaDE	1.0.5	Microarray meta-analysis for differentially expressed gene detection	√		√		√	√			√	√	MetaDE.pvalue
metaMA	3.1.2	Meta-analysis for MicroArrays	√				√	√			√		directEScombi
MetaPCA	0.1.4	MetaPCA: Meta-analysis in the dimension reduction of genomic data			√	√						√	MetaPCA
MetaQC	0.1.13	MetaQC: Objective quality control and inclusion/exclusion criteria for genomic meta-analysis			√	√					√	√	MetaQC
metaRNASeq	1.0.2	Meta-analysis of RNA-seq data	√				√				√		fishercomb
MetaSKAT	0.4	Meta-analysis for SNP-set (Sequence) Kernel Association Test	√		√		√	√					MetaSKAT_wZ

(continued)

TABLE 1. *(continued)*

Package	Version	Title	Effect Size	Power	Missing Data	Dependent Effects	Fixed Effect	Random Effects	Moderator Analyses	Publication Bias	Sensitivity Analysis	Creates Plots	Primary Function
`MultiMeta`	0.1	Meta-analysis of multivariate genome wide association studies						√	√			√	`multi_meta`
`seqMeta`	1.5	Meta-analysis of region-based tests of rare DNA variants	√				√	√				√	`burdenMeta`
`skatMeta`	1.4.3	Efficient meta-analysis for the SKAT test	√			√		√				√	`singlesnpMeta`
Multivariate meta-analysis													
`dosresmeta`	1.3.0	Performing multivariate dose–response meta-analysis	√			√	√	√				√	`dosresmeta`
`ecoreg`	0.2	Ecological regression using aggregate and individual data	√				√	√	√	√			`eco`
`metagen`	1	Inference in meta-analysis and meta-regression	√			√		√	√			√	`metareg`
`metaLik`	0.41.0	Likelihood inference in meta-analysis and meta-regression models						√	√				`metaLik`
`metaSEM`	0.9-2	Conducting univariate and multivariate meta-regression with structural equation modeling	√			√	√	√	√	√		√	`meta`
`metatest`	1.0-4	Fit and test meta-regression models						√	√				`metatest`
`mmeta`	2.2	Multivariate meta-analysis	√			√		√				√	`multipletables`
`mvmeta`	0.4.7	Multivariate and univariate meta-analysis and meta-regression	√		√	√	√	√	√			√	`mvmeta`
`mvtmeta`	1	Multivariate meta-analysis	√			√	√	√					`mvrmeta_re`
`robumeta`	1.6	Robust variance meta-regression				√		√	√		√	√	`robu`

(continued)

TABLE 1. *(continued)*

Package	Version	Title	Effect Size	Power	Missing Data	Dependent Effects	Fixed Effect	Random Effects	Moderator Analyses	Publication Bias	Sensitivity Analysis	Creates Plots	Primary Function
Diagnostic meta-analysis													
CopulaREMADA	0.9	Copula mixed effect models for bivariate and trivariate meta-analysis of diagnostic test accuracy studies	√					√	√			√	CopulaREMADA.norm
HSROC	2.1.8	Meta-analysis of diagnostic test accuracy when reference test is imperfect	√			√		√			√	√	HSROC
mada	0.5.7	Meta-analysis of diagnostic accuracy	√			√	√	√	√		√	√	madauni
metamisc	0.1.1	Diagnostic and prognostic meta-analysis	√		√	√	√	√				√	riley
Metatron	0.1-1	Meta-analysis for classification data and correction to imperfect reference	√			√		√	√		√		fit.bivar
Specific meta-analysis packages													
compute.es	0.2-4	Compute effect sizes	√								√		compute.es
CRTSize	1	Sample size estimation functions for cluster randomized trials		√			√						fixedMetaAnalRROR
exactmeta	1.0-2	Exact fixed effect meta-analysis	√				√						meta.exact
extfunnel	1.3	Additional funnel plot augmentations										√	extfunnel
forestplot	1.1	Advanced forest plot using "grid" graphics										√	forestplot
ipdmeta	2.4	Tools for subgroup analyses with multiple trial data using aggregate statistics	√	√			√	√	√			√	ipdlme

(continued)

TABLE 1. *(continued)*

Package	Version	Title	Effect Size	Power	Missing Data	Dependent Effects	Fixed Effect	Random Effects	Moderator Analyses	Publication Bias	Sensitivity Analysis	Creates Plots	Primary Function
`metap`	0.6	Meta-analysis of significance values					√					√	`metap`
`metaplot`	1.0	Meta-analysis forest plots										√	`drawmeta`
`SCMA`	1.1.1	Single-case meta-analysis	√										`combine`
Network meta-analysis													
`gemtc`	0.6-1	Network meta-analysis using Bayesian methods				√	√	√				√	`mtc.model`
`netmeta`	0.7-0	Network meta-analysis using frequentist methods	√			√	√	√			√	√	`netmeta`
`pcnetmeta`	2.1	Patient-centered network meta-analysis	√			√	√	√				√	`nma.ab`
Assessment of bias													
`metasens`	0.2-0	Advanced statistical methods to model and adjust for bias in meta-analysis	√				√	√		√	√	√	`copas`
`PubBias`	1	Performs simulation study to look for publication bias								√		√	`plot_chase_observed_expected`
`SAMURAI`	1.2.1	Sensitivity analysis of a meta-analysis with unpublished but registered analytical investigations	√				√	√		√		√	`forestsens`
`selectMeta`	1.0.7	Estimation of weight functions in meta-analysis						√		√		√	`IyenGreen`
`WeightFunctionModel`	1.0	The Vevea and Hedges weight-function model for publication bias				√	√		√		√	√	`n/a`

(continued)

TABLE 1. *(continued)*

Package	Version	Title	Effect Size	Power	Missing Data	Dependent Effects	Fixed Effect	Random Effects	Moderator Analyses	Publication Bias	Sensitivity Analysis	Creates Plots	Primary Function
Bayesian meta-analysis													
`bamdit`	1.1-1	Bayesian meta-analysis of diagnostic test data	√					√				√	`metadiag`
`bspmma`	0.1-1	Bayesian semiparametric models for meta-analysis						√				√	`dirichlet.c`
Graphical user interface													
`MAVIS`	1.1	Meta-analysis via Shiny	√		√	√	√	√	√	√	√	√	
`RcmdrPlugin.MA`	0.0-2	Graphical user interface for conducting meta-analyses in R	√		√	√	√	√	√	√	√	√	

Note. $N = 63$; package = package name; version = version number coded (may not be latest version); primary function = name of primary function in package; √ = package includes this functionality.

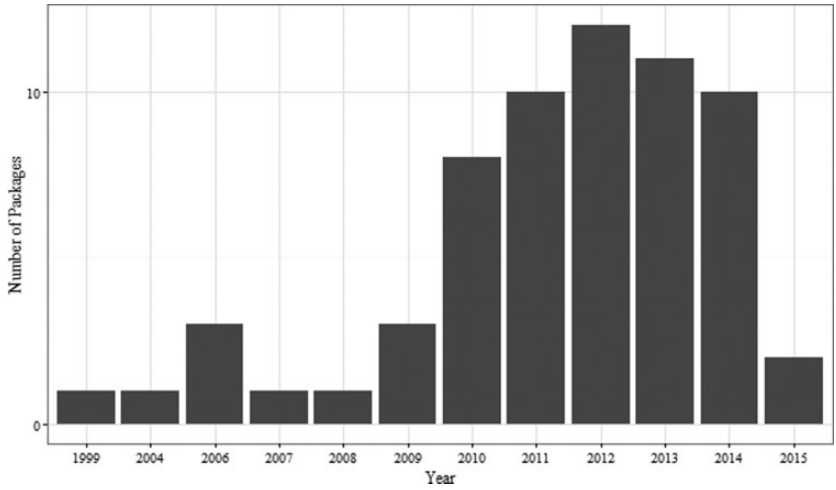

FIGURE 1. *Distribution of R meta-analytic package creation year.*

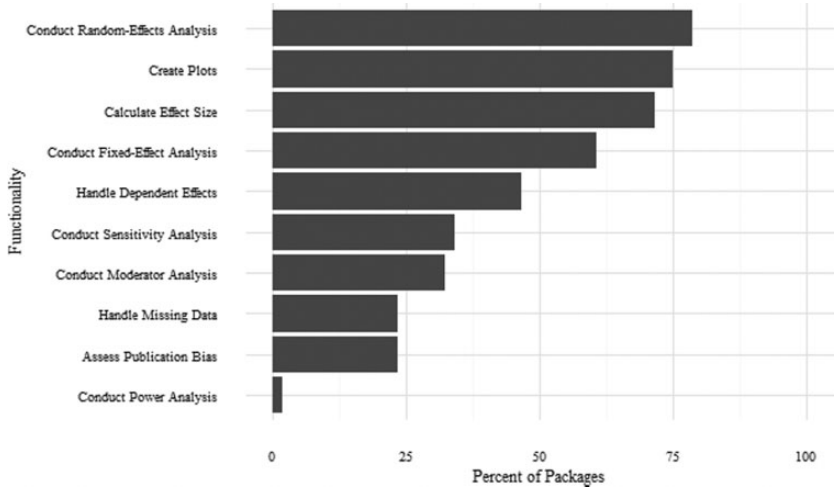

FIGURE 2. *Functionality across all R meta-analytic packages.*

render the plots. Conversely, few packages have the ability to address missing data (20.6%), conduct moderator analyses (31.8%), handle dependent effect sizes (44.4%), or assess publication bias (22.2%). Finally, a very small percentage can conduct a power analysis (3.17%). Figure 2 summarizes the packages across these various categories.

The packages also vary widely in terms of their use. The top meta-analysis package downloaded as of March 2016 was `epiR` (Stevenson, 2015), averaging 112.74 downloads per day ($SD = 26.06$). Across the entire data set, the average package was downloaded 8.56 times ($SD = 11.19$). Compared to popular R packages, however, meta-analytic packages are downloaded relatively infrequently. For example, the popular graphical analysis package `ggplot2` (Wickham, 2009) was downloaded an average of 8,467.13 times per day in March 2016.

4. Description of Meta-Analytic R Packages by Categories

4.1. General Meta-Analytic Packages

We consider 11 of the 63 total packages "general" meta-analysis packages (Table 1). The packages in this group do not focus on a specific topic area or function. Instead, the package authors intend for the packages to house many, if not all, traditional meta-analytic procedures. These 11 packages, therefore, should be considered the most appropriate for applied researchers who are interested in conducting a basic meta-analysis. Most of these packages can produce a fixed-effect or random-effects weighted mean, for instance, with only a few lines of simple code.

Arguably the most popular and pervasive package on this list is the package mentioned previously, `metafor` (Viechtbauer, 2015). The `metafor` package has many functions, allowing the user to conduct all parts of a meta-analysis, from calculating effect sizes to estimating a random-effects model, and plotting the results of the synthesis. The package will also address missing data using listwise deletion, handle hierarchically dependent effect sizes with multilevel modeling, conduct moderator and sensitivity analyses, and assess publication biases.

Along with the `metafor` package, several other general packages offer comprehensive functionality. The `MAc` (Del Re & Hoyt, 2015a) and `MAd` (Del Re & Hoyt, 2015b) are designed to estimate main effects with correlation and standardized mean difference effect sizes, respectively. They therefore offer specific corrections to effect sizes not available in the `metafor` package, for example. Similarly, the `metacor` (Laliberté, 2015) package focuses on correlation effect sizes but does not handle dependent effect sizes, and the documentation for this package is relatively sparse compared to the documentation for `MAc`. The `meta` (Schwarzer, 2015) and `rmeta` (Lumley, 2012) packages also offer comprehensive meta-analytic functionality but do not handle dependent effect sizes. Yang, Shi, and Xie's (2016) `gmeta` package includes multiple meta-analytic functions and will handle dependent effect sizes. The `rmeta` package, published in 1999, is the oldest known R package for meta-analysis (R Core Team, 2015).

A few of the general packages maintain multipurpose functionality but are domain-specific. For example, `epiR` (Stevenson, 2015) is the most popular meta-analysis package and focuses on epidemiology meta-analytic research. Another

example is the psychometric (Fletcher, 2015) package that is designed for ''applied psychometric theory,'' for instance, validity generalization, and includes functions to produce meta-analytic models as well; the metaplus package conducts robust meta-regression along with many other functions (Beath, 2015).

4.2. Genome Meta-Analytic Packages

The largest grouping of packages in our database constitutes the field of genomic and microarray meta-analysis ($n = 16$). All of these packages focus on meta-analysis for this specific content area. They are not suitable for general educational or social science meta-analyses.

Five packages focus on tools for meta-analysis of microarray data: MADAM (Kugler, Mueller, Ecker, & Graber, 2015), MAMA (Ihnatova, 2015), metaARRAY (Ghosh & Choi, 2015), MetaDE, (Wang, Li, & Tseng, 2015), and metaMA (Marot, 2015). The MAMA package can calculate effect sizes, run fixed- or random-effects models, and will create some plots. The MAMA package depends on the R packages GeneMeta (Lusa, Gentleman, & Ruschhaupt, 2015), metaMA, and metaARRAY. The metaARRAY package is used for large-scale meta-analysis of microarray data in a Bayesian framework and includes an author-written demonstration of the package capabilities using publicly available data (Ghosh & Choi, 2015). The MetaDE package gives users the option to implement 12 types of meta-analysis for differential gene expression. Users can also choose which test statistic to implement based on the outcome variable and choose corrections for one-sided tests. The MetaDE package appears to be the only microarray meta-analysis package that addresses missing data: The program will impute missing data for any gene with less than 30% missing data and will ignore genes with more than 30% data missing. The metaMA package looks for differentially expressed genes by combining either *p* values or *t*-test statistics in both paired and unpaired data. There is also a demonstration vignette using publicly available data for this package (Marot & Bruyère, 2015). Marot, Jaffrezic, and Rau's (2015) metaRNAseq has similar functionality that can be used with RNA sequencing experiments.

The other packages in this category conduct a variety of specific tasks. Kang and Tseng's (2015a) metaPCA conducts factor reduction procedures, while Kang and Tseng's (2015b) other program, metaQC, seeks to improve the inclusion and exclusion criteria of data. Lee's (2015) metaSKAT program can be used to ''carry out gene-based meta-analysis for rare variants'' (p. 1). Voorman, Brody, and Lumley's (2015) skatMETA and Voorman, Brody, Chen, and Lumley's (2015) seqMETA have similar functionalities. The metABEL (Struchalin & Aulchenko, 2015) package is designed to conduct meta-analyses of ''genome-wide association scans between quantitative or binary trials and SNPs'' (p. 1). Two packages, MultiMETA (Vuckovic, 2015) and EasyStrata (Winker, 2015), work with genome-

wide data specifically. Zhao's (2015) `gap` package will conduct general genomic sequencing meta-analysis processes.

4.3. Multivariate Meta-Analytic Packages

A group of packages that may be useful for both applied and advanced researchers is the multivariate meta-analysis category (Table 1). The packages ($n = 10$) that constitute this grouping share multivariate statistical processes that allow for the use of multiple covariates or outcomes.

This category can be divided further into packages that can handle dependent effect sizes or those that cannot. Within the multivariate package category, seven packages can handle dependent effect sizes. Three of those packages perform general and traditional multivariate meta-analysis, `mmeta` (Luo, Chen, Su, & Chu, 2015), `mvmeta` (Gasparrini, 2015), and `mvtmeta` (Chen, 2015), for analyses where the covariation among the dependent effect sizes is known or can be estimated. The `mvmeta` package, for example, uses the covariation among the dependent effect sizes to adjust outcome effect sizes and can be used to fit models with multiple time points, network meta-analytic models, or diagnostic testing. The `mvmeta` package can also conduct simulations. The `ecoreg` package may be used to include individual participant data as well (Jackson, 2015).

Three other packages that account for the dependency among the effect sizes are `dosresmeta` (Crippa, 2015), `metaSEM` (Cheung, 2015), and `robumeta` (Fisher & Tipton, 2015). These packages do not require knowledge of the covariance among the effect sizes; yet, all provide unbiased estimates of standard errors. The `dosresmeta` package estimates covariances using methods proposed by Greenland and Longnecker (1992) and Hamling and colleagues (2008). The `robumeta` package estimates standard errors using a robust variance estimation technique, proposed by Hedges, Tipton, and Johnson (2010). The `metaSEM` package utilizes a structural equation modeling framework to account for within-study dependencies. There is also extensive instruction and examples available on the developer's website for the `metaSEM` package. These packages' greatest benefits are that multiple dependent effect sizes can be synthesized simultaneously without knowing the exact covariance among the effect sizes. Although these packages include plotting functions, the `robumeta` (Fisher & Tipton, 2015) package will produce traditional forest plots grouping the effect sizes by study, which is unique to this package.

The second group of packages focuses on conducting meta-regression models but do not handle dependent effect sizes ($n = 3$), so analysts must choose only one effect size per study to include in the model if using these packages: `metagen` (Möbius, 2015), `metaLik` (Guolo & Varin, 2015), and `metatest` (Huizenga & Visser, 2015). The first package, `metagen`, has the capacity to conduct several different types of inference-based meta-regression models. The second package,

metaLik, includes functionality to conduct "hypothesis testing on a scalar component of the fixed-effect vector . . . using the signed log-likelihood ratio test" (p. 9). Simulation functions are also available. The third package, metatest, produces an array of test statistics for various parameters in the estimation of meta-regression models.

4.4. Diagnostic Test Accuracy Meta-Analytic Packages

Although several other package authors specifically state that diagnostic test accuracy meta-analyses may be performed using their package (e.g., metafor and mvmeta), five packages are designed solely to conduct diagnostic meta-analyses (Table 1). Three of these packages are intended to conduct diagnostic meta-analyses generally, while two are useful to address specific data issues. The package mada (Doebler, 2015) provides functionality to estimate fixed- and random-effects analyses, conduct moderator and sensitivity analyses, and plot meta-analytic results and receiver operating characteristics (ROC) curves. The package, metamisc (Debray, 2015), is also designed to handle prognostic testing, and it can apply frequentist or Bayesian approaches and handle dependencies within studies; however, it does not have functionality to conduct moderator or sensitivity analyses. The CopulaREMADA package (Nikoloulopoulos, 2015) uses diagnostic test accuracy data for bivariate and trivariate meta-analyses.

The HSROC (Schiller & Dendukuri, 2015) and Metatron (Huang, 2015) packages conduct diagnostic test accuracy meta-analyses when the reference test is not ideal. The Metatron (Huang, 2015) package allows users to conduct traditional diagnostic meta-analyses while also estimating the impact of specific studies on the averaged point estimates. The HSROC package, on the other hand, allows for the estimation of both within- and between-study variability using a hierarchical modeling technique. Therefore, this package is robust to violations of standard errors due to hierarchical clustering.

4.5. Network Meta-Analysis Packages

There are three network meta-analysis packages: gemtc (van Valkenhoef & Kuiper, 2014), pcnetmeta (Lin, Zhang, & Chu, 2015), and netmeta (Rücker, Schwarzer, Krahn, & Konig, 2015). One package utilizes a generic Bayesian approach (gemtc) and the other uses a Bayesian approach specifically for patient-centered network meta-analysis (pcnetmeta). The final package in this category, netmeta, conducts network meta-analysis using a frequentist approach under the graph-theoretical method (Rücker, 2012). All three packages have multiple plotting capabilities, including the forest plot (labeled the blobbogram in gemtc), network graph, and net heat plot. The pcnetmeta package relies on Just Another Gibbs Sampler for parameter estimation, which must be downloaded separately (http://mcmc-jags.sourceforge.net). For a larger discussion on network meta-

analysis packages, as well as a tutorial and explanation of the functionality, see Neupane et al.'s (2014) report on these packages.

4.6. Bayesian Meta-Analysis Packages

Two packages were specific to Bayesian approaches to meta-analysis. The `bamdit` (Verde & Sykosch, 2015) package enables Bayesian random-effects modeling for diagnostic test data and can calculate effect sizes and produce the Bayesian Summary ROC curve. It also has the capability, through `ggplot2` (Wickham, 2009), to plot the true- versus the false-positive rates of each study in the meta-analysis and to show included subgroups using different colors. The second package using a Bayesian approach is `bspmma` (Burr, 2015). This package estimates nonparametric and semiparametic random effects meta-analyses but has limited plotting capabilities.

4.7. Assessment of Bias Packages

Five packages are used primarily for detecting bias, four of which specifically assess the potential of publication bias in meta-analyses. The four packages addressing publication bias are `PubBias` (Thornley, 2015), `SAMURAI` (Noory 2015), `selectMeta` (Rufibach, 2015), and `WeightFunctionModel` (Vevea & Coburn, 2015). The `PubBias` package assesses whether the observed number of positive effects in the meta-analysis is greater than what is expected using Fisher's exact test. The `SAMURAI` package imputes missing effect sizes for binary and continuous outcomes from unpublished studies and produces a forest or funnel plot with the outcome. The `selectmeta` package addresses publication bias by modeling potential bias through the use of nonparametric and parametric weight functions. The `WeightFunctionModel` runs through R using a web-based user interface (https://vevealab.shinyapps.io/WeightFunctionModel/). To run the program, the user completes a series of questions about the data, selects the variables (including a moderator if necessary), and sets the desired *p* value cut point. The program will then compute the Vevea and Hedges (1995, 1996) weight function model for publication bias and produce an accompanying funnel plot.

The final package in this category, `metasens` (Schwarzer, Carpenter, & Rücker, 2015a), is a broader package for assessing potential sources of bias. Through this package, a limit meta-analysis can be conducted and the package will also produce the following plots to assess potential bias: funnel, contour, treatment effect, and *p* value for residual selection bias.

4.8. Packages With Specific Functionality

There are nine packages that limit functionality to particular aspects of conducting a meta-analysis or to a specific type of data. One package, `compute.es`

(Del Re, 2015a), is primarily useful for calculating six popular effect sizes (Cohen's *d*, Hedges' *g*, *r*, Fisher's *z*, odds ratio, and log odds ratio) and their variances, confidence intervals, and *p* values. The `MAc` package in R and the `MAVIS` GUI uses this package. Another package, `metap` (Dewey, 2015), can combine *p* values in a number of different ways including the logit, mean *p*, sum of logs, sum of *p*, sum of *z*, vote counting, and Wilkinson's methods.

Three packages offer extensions to existing meta-analytic plotting options: (1) `extfunnel` (Langan, Sutton, Higgins, & Gegory, 2015), which can be used to augment funnel plots with statistical significance or heterogeneity contours; (2) `forestplot` (Gordon & Lumley, 2015), which uses grid graphics for more advanced forest plot options including outside of meta-analyses; and (3) `metaplot` which uses output from R packages `meta`, `metafor`, and/or `rmeta` (Murrell, Scott, & Lu, 2014). Finally, three packages focus on specific types of data for meta-analysis. The package `exactmeta` (Yu & Tian, 2015) conducts fixed-effect meta-analyses for rare event data, the `ipdmeta` (Kovalchik, 2015) package conducts meta-analyses of individual patient data, and the `SCMA` (Bulte, 2015) package enables meta-analysis of single-case data.

Of particular interest to meta-analysts, given the few packages that include this functionality, the package `CRTsize` (Rotondi, 2015) includes functionality to conduct meta-analytic power estimates.

4.9. GUI Applications

Two of the meta-analysis programs feature GUIs: `MAVIS` (Hamilton, Mizumoto, Aydin, & Coburn, 2015) and `RcmdrPlugin.MA` (Del Re, 2015b). The benefit of using a GUI in R is that the user does not need to know how to code or use R beyond a rudimentary understanding. The `MAVIS` package has the feel of SPSS, where the user interacts with R through a series of drop-down menus. The package allows for fixed- and random-effects models, conducts basic moderator analyses, and has plotting capabilities. The `RcmdrPlugin.MA` is a supplement to the popular `Rcmdr` package (Fox et al., 2016). The user first downloads the `Rcmdr` GUI and then adds the meta-analysis package to it. Using the existing user interface, the user may choose meta-analytic options from a built-in drop-down menu. For individuals who do not want to learn the R programming languages, either of these options provides simple to use but high-level meta-analytic functionality.

5. Tutorial of Two Meta-Analysis Packages

As a follow-up to the discussion on the packages, we now provide a brief tutorial on two of the popular meta-analytic packages, `metafor` (Viechtbauer, 2015) and `robumeta` (Fisher & Tipton, 2015). The packages' help files provide many more examples and illustrate greater functionality, and a growing

community of active users may be found on sites such as Stack Overflow (Stack Exchange, 2016). From our experience, however, it can be difficult for new R users to learn new R packages. Therefore, the purpose of this section is to provide enough detail to the neophyte R user and meta-analyst such that the basic meta-analytic procedures may be conducted. In addition, we illustrate the advantage of using robust variance estimation over traditional meta-analytic models. Robust variance estimation allows the meta-analyst to synthesize all available effect sizes in one model, even when more than one effect size is available in a particular study. Until recently, most meta-analysts either (a) choose only one effect size per study to synthesize, (b) averaged the multiple effect sizes within each study prior to synthesizing across studies, or (c) split the meta-analysis into smaller, outcome-specific meta-analyses. Users of robust variance estimation do not face any of these problems.

The context for these examples is from a recent systematic review and meta-analysis on prevention programs designed to decrease teen dating violence in middle and high school students (De La Rue, Polanin, Espelage, & Pigott, 2016). The outcome of interest for this tutorial is teen dating violence knowledge, and the effect sizes are coded such that a negative effect size indicates a beneficial treatment effect (i.e., negative effect sizes represent a reduction in teen dating violence). The data set includes 13 independent samples and 25 effect sizes. Table 2 provides all the information needed to run R code. In addition to the effect size (labeled column ''g''), the example data set includes the effect size variance (var), study ID (Study.ID), effect size ID (ES.ID), author last name (Author), and whether or not the effect size is from a follow-up time point (Follow.Up). Three moderator variables are also listed: date of publication (DOP), whether the study used random assignment (Random), and the percentage of males included in the intervention group (Perc.Males). The results are presented for demonstration only and audiences interested in the full results should consult the publication. We include a tutorial of the code prior to each section and provide information within the code using the ''#'' to indicate a comment.

5.1. Package Installation and Loading

Start by clearing the R environment and loading the meta-analysis packages. If this is the first time using `metafor` or `robumeta`, the packages will need to be installed. To install the packages, type into the R console: `install.packages("metafor")`. This will initiate installation. The last piece of code (`sessionInfo`) lists the R and package versions the following code was run in. If running the code is a problem, use these R and package versions as a way to troubleshoot.

TABLE 2.
Teen Dating Violence Meta-Analysis Data Set

Study.ID	ES.ID	DOP	*g*	var	Follow.Up	Random	Perc.Males
1	1	1998	0.010	0.015	0	1	48.9
1	2	1998	0.010	0.015	1	1	48.9
2	3	2006	−0.707	0.002	0	1	48.8
2	4	2006	−0.349	0.002	1	1	48.8
3	5	1991	−0.279	0.007	0	0	NA
3	6	1991	−0.211	0.007	0	0	NA
4	7	1996	−0.286	0.009	0	0	NA
4	8	1996	0.003	0.009	1	0	NA
5	9	1997	−0.272	0.009	0	1	49.3
6	10	2002	−0.021	0.007	0	0	45.03
6	11	2002	−0.047	0.007	1	0	45.03
6	12	2002	0.348	0.007	0	0	45.03
6	13	2002	0.072	0.007	1	0	45.03
6	14	2002	0.355	0.012	0	0	45.91
6	15	2002	0.036	0.012	1	0	45.91
6	16	2002	0.389	0.012	0	0	45.91
6	17	2002	−1.739	0.016	1	0	45.91
7	18	2000	−0.121	0.001	0	0	50.5
8	19	2000	−0.492	0.014	0	1	42.8
9	20	2010	−0.206	0.012	0	1	52
9	21	2010	−0.194	0.010	1	1	52
10	22	2012	−0.165	0.011	0	1	54.2
10	23	2012	−0.119	0.006	1	1	54.2
11	24	2001	−1.233	0.199	0	0	45.8
11	25	2001	−1.425	0.209	1	0	45.8

Note. DOP = date of publication.

```
rm(list = ls())
 }]
library(metafor)

## Loading required package: Matrix
## Loading 'metafor' package (version 1.9-8). For an overview
## and introduction to the package please type: help(metafor).

library(robumeta)

## Loading required package: grid

sessionInfo()

## R version 3.2.2 (2015-08-14)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
```

```
## Running under: Windows 7 x64 (build 7601) Service Pack 1
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] grid   stats   graphics   grDevices   utils   datasets   methods
## [8] base
##
## other attached packages:
## [1] robumeta_1.6 metafor_1.9-8 Matrix_1.2-2
##
## loaded via a namespace (and not attached):
## [1] magrittr_1.5   tools_3.2.2       htmltools_0.2.6  yaml_2.1.13
## [5] stringi_1.0-1  rmarkdown_0.8.1   knitr_1.11       stringr_1.0.0
## [9] digest_0.6.8   lattice_0.20-33   evaluate_0.8
```

5.2. Upload and Prepare Data

Next upload the data set (Data_ TDV.csv) using the "`read.csv`" function. The "head" and "str" functions allow the user to visualize the data set in an efficient manner. After ensuring that the data set is correctly uploaded, we split the data set into two parts using the "Follow.Up" variable. The rows coded as 0 are for "posttest" measures, while the 1 rows are follow-up time points. It is often not appropriate to synthesize posttest and follow-up measures simultaneously and thus here we run the meta-analyses separately by follow-up time point.

```
dat <- read.csv("Data_TDV.csv", header = T)
head(dat)
```

```
##   Study.ID ES.ID          Author  DOP Follow.Up          g         var
## 1      105     1   Foshee, V. A. 1998         0  0.0100000 0.014621856
## 2      105     2   Foshee, V. A. 1998         1  0.0100000 0.014621856
## 3      110     3    Jaycox, L.H. 2006         0 -0.7074087 0.001864498
## 4      110     4    Jaycox, L.H. 2006         1 -0.3487226 0.001781894
## 5      111     5 Jones, L.E. (1) 1991         0 -0.2792240 0.007212470
## 6      112     6 Jones, L.E. (2) 1991         0 -0.2107261 0.007387867
## Random Perc.Males
## 1      1         48.9
```

```
## 2     1     48.9
## 3     1     48.8
## 4     1     48.8
## 5     0       NA
## 6     0       NA
```

```
str(dat)
```

```
## 'data.frame':  25 obs. of 9 variables:
##  $ Study.ID  : int 105 105 110 110 111 112 113 113 114 117...
##  $ ES.ID     : int 1 2 3 4 5 6 7 8 9 10...
##  $ Author    : Factor w/ 12 levels 'Foshee, V. A.",..: 1 1 2 2 3 4 5 5 6 7...
##  $ DOP       : int 1998 1998 2006 2006 1991 1991 1996 1996 1997 2002...
##  $ Follow.Up : int 0 1 0 1 0 0 0 1 0 0...
##  $ g         : num 0.01 0.01 -0.707 -0.349 -0.279...
##  $ var       : num 0.01462 0.01462 0.00186 0.00178 0.00721...
##  $ Random    : int 1 1 1 1 0 0 0 0 1 0...
##  $ Perc.Males: num 48.9 48.9 48.8 48.8 NA...
```

```
dat.0 <- subset(dat, Follow.Up == "0")
dat.1 <- subset(dat, Follow.Up == "1")
```

5.3. Prepare the Data Set for a Traditional Meta-Analysis

This section details the steps necessary to conduct a "traditional" meta-analysis where all effect sizes are independent. In Section 5.5, we illustrate the steps necessary to conduct the state-of-the-art meta-analysis technique that utilizes robust variance estimation. We remind readers that we recommend using the newer technique when possible.

The first step in conducting a traditional meta-analysis is to ensure that all effect sizes are independent. When multiple effect sizes from the same sample are represented in a single study, one option to ensure independence is to average the effect sizes and their variances. We start by subsetting the rows that have multiple effect sizes per study (rows 4–5 and 6–7 of the data set). Within each study, we average the effect sizes and variances and create an object to represent each statistic (e.g., "avg.g.117"). These averages then replace the statistics listed in the original effect size and variance rows.

```
#create a small dataset with the studies that include multiple effect sizes
dat.1.117 <- dat.1[c(4,5),]
dat.1.118 <- dat.1[c(6,7),]

#average the effects within each study
avg.g.117 <- mean(dat.1.117$g)
avg.var.117 <- mean(dat.1.117$var)
```

```
avg.g.118 <- mean(dat.1.118$g)
avg.var.118 <- mean(dat.1.118$var)

#replace the multiple effect sizes with the average effect size
dat.1.117[1,6] <- avg.g.117
dat.1.117[1,7] <- avg.var.117

dat.1.118[1,6] <- avg.g.118
dat.1.118[1,7] <- avg.var.118
```

Finally, the rows with the new effect sizes and variances are combined with the original data set (rbind(dat.1, dat.1.117, dat.1.118)) and the old rows with unused data are removed (dat.1[-c(4,5,6,7,12,14),]). What remains is a meta-analytic data set that has independent effect sizes and variances.

```
dat.1 <- rbind(dat.1, dat.1.117, dat.1.118)
dat.1 <- dat.1[-c(4,5,6,7,12,14),]
```

5.4. Conduct a Traditional Meta-Analysis Using `metafor`

With the data set finalized, it is now possible to conduct a traditional meta-analysis. Four distinct analyses are conducted below. The first (run.1) is a simple meta-analysis with no moderators. The "yi" and "vi" arguments are for the effect size and variances, respectively. The "data" argument tells R which data set to use, and the "method" argument is reserved for the type of meta-analytic model to run. In this case, we estimate a random-effects model using the maximum likelihood estimator (ML). After running the code, we type "summary(run.1)" to display the output.

The output lists all of the important and relevant information about the conducted meta-analysis. Under the "Model Results" heading, we can see the average effect (estimate), the standard error for the average effect (se), the *z*-statistic testing the statistical significance of the average effect (zval), the *p* value associated with the *z*-statistic (pval), and the 95% confidence interval's lower and upper bounds ("ci.lb" and "ci.ub," respectively). The model's heterogeneity information is found in "Test for Heterogeneity." In this example, the results indicate a statistically significant mean effect size, but substantial heterogeneity in effects; thus, additional moderator tests may be warranted to explain some of this heterogeneity in effects.

```
run.1 <- rma.uni(yi = g, #effect size
                 vi = var, #variance
                 data = dat.1, #dataset
                 method = "ML" #model estimation )
```

```
summary(run.1)
##
## Random-Effects Model (k = 8; tau^2 estimator: ML)
##
##  logLik  deviance  AIC     BIC      AICc
##  -3.8449 28.7496   11.6897 11.8486  14.0897
##
## tau^2 (estimated amount of total heterogeneity): 0.0874 (SE = 0.0510)
## tau (square root of estimated tau^2 value):      0.2957
## I^2 (total heterogeneity / total variability):   91.84%
## H^2 (total variability / sampling variability):  12.26
##
## Test for Heterogeneity:
## Q(df = 7) = 62.4666, p-val < .0001
##
## Model Results:
##
## estimate      se    zval   pval   ci.lb   ci.ub
##  -0.2644  0.1146 -2.3072 0.0210 -0.4890 -0.0398 *
##
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There are various options available to conduct moderator analyses, and the second, third, and fourth models each run a different type of moderator analysis. The second model runs a one-way analysis of variance type moderator analysis using the "`mods`" argument. The "−1" at the end of the statement tells R to remove the intercept and therefore the coefficients represented in the model are the meta-analytic average effect sizes for each level (i.e., random and non-random). The "Test of Moderators" provides the Q-between statistic, which indicates that the difference between the groups was statistically significant (Q-between = 6.29, p = .04).

```
run.2 <- rma.uni(yi = g,
                 vi = var,
                 data = dat.1,
                 method = "ML",
                 mods =~ factor(Random) - 1 #moderator argument)
summary(run.2)
```

```
##
## Mixed-Effects Model (k = 8; tau^2 estimator: ML)
##
##   logLik   deviance    AIC     BIC    AICc
##   -3.4152  27.8903 12.8304 13.0687 18.8304
##
```

```
## tau^2 (estimated amount of residual heterogeneity):  0.0847 (SE = 0.0495)
## tau (square root of estimated tau^2 value):          0.2910
## I^2 (residual heterogeneity / unaccounted variability): 90.97%
## H^2 (unaccounted variability / sampling variability): 11.07
##
## Test for Residual Heterogeneity:
## QE(df = 6) = 61.7007, p-val < .0001
##
## Test of Moderators (coefficient(s) 1,2):
## QM(df = 2) = 6.2936, p-val = 0.0430
##
## Model Results:
##
##                 estimate     se    zval   pval   ci.lb   ci.ub
## factor(Random)0  -0.3797 0.1688 -2.2495 0.0245 -0.7106 -0.0489 *
## factor(Random)1  -0.1689 0.1521 -1.1105 0.2668 -0.4669
0.1292
##
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The third and fourth models are subgroup type analyses. In these analyses, the data set is split (or subset) into the two groups that represent random and nonrandom assignment. We again use the "`summary`" function to see the results. Studies that use a nonrandom assignment procedure have a larger average effect, but the effect size is not statistically significant ($p = .09$). Studies that use random assignment have a smaller effect, and the effect size is statistically significant ($p = .01$). We should note that the meta-analytic average effect sizes differ from those in Model 2 because Models 3 and 4 do not include the between-study variance, whereas Model 2 includes this variance component (see Borenstein, Hedges, Higgins, & Rothstein, 2010, for more information about the difference between moderator and subgroup analyses).

```
run.3 <- rma.uni(yi = g,
                 vi = var,
                 data = dat.1,
                 method = "ML",
                 subset = (Random == "0") #subgroup argument)

run.4 <- rma.uni(yi = g,
                 vi = var,
                 data = dat.1,
                 method = "ML",
                 subset = (Random == "1") #subgroup argument)
```

```
summary(run.3)
##
## Random-Effects Model (k = 4; tau^2 estimator: ML)
##
##  logLik  deviance    AIC     BIC   AICc
## -3.4159  14.9449  10.8317 9.6043 22.8317
##
## tau^2 (estimated amount of total heterogeneity): 0.2358 (SE = 0.1913)
## tau (square root of estimated tau^2 value):      0.4856
## I^2 (total heterogeneity / total variability):   94.35%
## H^2 (total variability / sampling variability):  17.70
##
## Test for Heterogeneity:
## Q(df = 3) = 48.4943, p-val < .0001
##
## Model Results:
##
##  estimate    se    zval   pval   ci.lb  ci.ub
##  -0.4533 0.2631 -1.7231 0.0849 -0.9690 0.0623

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

```
summary(run.4)
##
## Random-Effects Model (k = 4; tau^2 estimator: ML)
##
## logLik   deviance     AIC      BIC   AICc
## 2.3953   8.1562    -0.7906 -2.0180 11.2094
##
## tau^2 (estimated amount of total heterogeneity): 0.0114 (SE = 0.0125)
## tau (square root of estimated tau^2 value):      0.1069
## I^2 (total heterogeneity / total variability):  65.63%
## H^2 (total variability / sampling variability):  2.91
##
## Test for Heterogeneity:
## Q(df = 3) = 13.2064, p-val = 0.0042
##
## Model Results:
##
##    estimate      se    zval   pval   ci.lb   ci.ub
##   -0.1921    0.0676 -2.8395 0.0045 -0.3246 -0.0595 **
##
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5.5. Conduct Robust Variance Estimation Meta-Analysis Using `robumeta`

We next turn to robust variance meta-analysis using the "dat.0" data set, which is the data set with posttest-only effect sizes. The "dat.0" data set was created in Section 5.2 using the "subset" function. Because we are using a robust variance model, we do not have to ensure independent effect sizes and therefore do not have to manipulate the data set as we did in Section 5.3. The model accounts for the dependence within each study by assuming a constant within-study correlation among the effect sizes. We can conduct sensitivity analyses to ensure that this correlation parameter does not significantly impact the results.

We start by running a meta-analysis model with no moderators. For the `robumeta` package, we must tell the package what the effect size "formula" is. This formula statement will later be used to test moderators. For now, the formula statement simply tells R which column is the effect size column (g) and to regress it on a column of 1s, hence the "formula = g ~ 1" statement. The effect size variance is stated through the "`var.eff.size`" statement, "`data`" shows the package where the data are located, and "`studynum`" tells the package which column has the study-level IDs. Should multiple effect sizes be available per study, the effect sizes from the same study should each have the same "Study.ID" number. The final statement, "`modelweights = 'CORR'`," tells the package that the dependence is due to multiple effect sizes per study sample. The other `modelweights` option, "`HIER`," is used when the effect sizes derive from some hierarchical cluster (e.g., author team, lab, or country), which is not applicable for this scenario. To see the output, we use the "`print`" statement. We can see that average effect size is statistically significant.

```
run.5 <- robu(formula = g ~ 1, #effect size
              var.eff.size = var, #variance
              data = dat.0, #dataset
              studynum = Study.ID, #study-level ID
              modelweights = "CORR" #RVE model estimation)

print(run.5)
```

```
## RVE: Correlated Effects Model with Small-Sample Corrections
##
## Model: g ~ 1
##
## Number of studies = 13
## Number of outcomes = 15 (min = 1, mean = 1.15, median = 1, max = 2)
## Rho = 0.8
## I.sq = 94.15285
## Tau.sq = 0.09484842
##
##             Estimate  StdErr t-value  dfs  P(|t|>) 95% CI.L  95% CI.U Sig
## 1 intercept -0.216    0.085   -2.54   11.6  0.0265  -0.402     -0.03  **
## ---
```

```
## Signif. codes: < .01 *** < .05 ** < .10 *
## ---
## Note: If df < 4, do not trust the results
```

To test whether the results are sensitive to the within-study correlation, we also run the "`sensitivity(run.5)`" statement. This function varies the within-study correlation across a range of plausible values between 0 and 1, and the output shows the model's results for each variation of the within-study correlation. For the current example, varying ρ across five values does not change the overall average ($g = -.2159$). We can be confident that the within-study correlation does not impact the results.

```
sensitivity(run.5)
```

```
## RVE: Correlated Effects Model with Small-Sample Corrections
## Model: g ~ 1
##
## Sensitivity Analysis
##
##                       Rho = 0 Rho = 0.2 Rho = 0.4 Rho = 0.6 Rho = 0.8
## intercept Coefficient -0.2159 -0.2159   -0.2159   -0.2159   -0.2159
##         Std. Error     0.0850  0.0850    0.0850    0.0850    0.0850
## Tau.sq    Estimate     0.0948  0.0948    0.0948    0.0948    0.0948
## Rho = 1
## -0.2159
##  0.0850
##  0.0949
```

The final two models, 6 and 7, run moderator analyses. Model 6 tests for the difference between random and nonrandom assignment. The model results indicate that the effect sizes do not differ significantly across these two groups ($p = .23$). The "Note" that reads "If $df < 4$, do not trust the results" refers to the degrees of freedom associated with the regression coefficients. Tanner-Smith et al. (2016) suggested that results with small degrees may be unstable and therefore caution is warranted.

```
run.6 <- robu(formula = g ~ factor(Random) #moderator argument,
              var.eff.size = var,
              data = dat.0,
              studynum = Study.ID,
              modelweights = "CORR")

print(run.6)
```

```
## RVE: Correlated Effects Model with Small-Sample Corrections
##
## Model: g ~ factor(Random)
```

```
##
## Number of studies = 13
## Number of outcomes = 15 (min = 1 , mean = 1.15 , median = 1 , max = 2 )
## Rho = 0.8
## I.sq = 90.30503
## Tau.sq = 0.06545406
##
##                  Estimate StdErr t-value  dfs P(|t|>) 95% CI.L 95% CI.U
## 1      intercept  -0.116  0.114   -1.01   5.47  0.354     -0.401   0.17
## 2 factor.Random.1 -0.200  0.158   -1.27  10.36  0.232     -0.550   0.15
##  Sig
## 1
## 2
## ---
## Signif. codes: < .01 *** < .05 ** < .10 *
## ---
## Note: If df < 4, do not trust the results
```

Model 7 is a multiple meta-regression model that includes the "random" binary variable and the "Perc.Males" continuous variable. Again, the results indicate that the moderators are not statistically significant predictors of effect size magnitude.

```
run.7 <- robu(formula = g ~ factor(Random) + scale(Perc.Males),
              var.eff.size = var,
              data = dat.0,
              studynum = Study.ID,
              modelweights = "CORR")

print(run.7)
## RVE: Correlated Effects Model with Small-Sample Corrections
##
## Model: g ~ factor(Random) + scale(Perc.Males)
##
## Number of studies = 10
## Number of outcomes = 12 (min = 1 , mean = 1.2 , median = 1 , max = 2 )
## Rho = 0.8
## I.sq = 93.06073
## Tau.sq = 0.113052
##
##                   Estimate StdErr t-value  dfs P(|t|>) 95% CI.L 95% CI.U
## 1   intercept      -0.00981  0.220 -0.0446  2.67  0.968   -0.763   0.743
## 2 factor.Random.1  -0.32827  0.236 -1.3890  3.89  0.239   -0.992   0.335
## 3 scale.Perc.Males. 0.05898  0.056  1.0533  2.52  0.383   -0.140   0.258
## Sig
## 1
## 2
```

```
## 3
## ---
## Signif. codes: < .01 *** < .05 ** < .10 *
## ---
## Note: If df < 4, do not trust the results
```

6. Discussion

The R statistical environment is a popular and useful tool for researchers and statisticians. Its popularity has grown steadily over the past decade, and with it, the growth of user-created and open-source R ''packages'' has provided increased functionality and utility. The meta-analytic community has contributed to this growth by adding many additional packages specific to meta-analysis or including meta-analysis functionality. The purpose of this article is to locate all publicly available meta-analytic R packages, describe their functionality and use, and provide a brief tutorial of two popular packages.

For general functionality, we recommend that the new user start with the ''general meta-analysis'' packages, and of those, the `metafor` (Viechtbauer, 2015) package provides many useful functions. This package is one of the most highly downloaded R packages for meta-analysis and is also one of the most flexible packages that is appropriate for researchers interested in conducting a meta-analysis. Numerous specialized packages are also available, including packages for conducting meta-analysis of diagnostic test accuracy studies, genome data, and conducting Bayesian meta-analyses.

6.1. Limitations of the Review

One of the limitations of the current study is that we only reviewed software for conducting the statistical technique of meta-analysis and did not review packages or other software that might be useful for conducting systematic reviews and extracting meta-analytic data. Meta-analyses are often a result of a systematic literature search, and this process involves extensive data management. R may be a useful tool in this process, but we are not aware of any R packages that address this issue. In an ideal world, a package to manage each step of a systematic review and meta-analysis might be useful. This may be an area for future development. In addition, given the nature of open-source programming, in the time since we finished searching for, collecting, and coding packages, it is possible that (a) more packages have been created and (b) the packages included in this review have been updated to include more (or less) functionality.

6.2. Future Functionality

From the results of this review, we have four recommendations for improving or expanding current R packages for meta-analysis. One of the limitations of using R for meta-analysis is that the knowledge needed for command line analysis can be burdensome for some researchers and often requires additional training relative to software with user-friendly GUIs. Thus, our first recommendation to enable increased functionality is for package authors to consider creating a GUI or, at the very least, to provide at least one demonstration vignette with their package. Some packages already provide excellent examples for their end users (e.g., `metafor`, `metaARRAY`, `metaMA`, and `metaSEM`). Although maintaining a separate website outside the CRAN repository as some package authors have done is not always feasible, an example demonstration can easily be stored in CRAN documentation.

Second, based on the results, it is clear that R packages lack functionality to conduct meta-analytic power estimates. Power analyses for the overall average effect are limited to only a few packages. Completely missing from these packages is functionality to conduct power analyses for meta-analytic moderator analyses. A package that includes this functionality will prove most helpful to the field. Third, we recommend that authors enhance the user-friendliness of meta-analysis graphing functions because refining graphs can be difficult for users. For example, a user-friendly GUI for traditional meta-analysis graphs such as forest plots, bubble plots, Galbraith plots, and funnel plots will ultimately help end users. Fourth and finally, package authors should attempt to consistently report key information about their package for the novice R user—that is, whether the user can calculate fixed- or random-effects model is important for any user to know when choosing an R package for meta-analysis. It was often difficult to locate what functionalities were available or how the packages implemented the procedures and this information should be included in all package documentation.

Finally, it is not our intention to be critical of any one package because all of these packages provide a beneficial service to the community. We recognize the difficulty in finding appropriate and easy-to-use functionality, through our own struggles, and provide this review article simply as a resource for current and future meta-analysts. We believe in the utility of meta-analysis and wish to see the community of users and analysts grow. Through continued refinement, meta-analytic R packages have the potential to provide the field with powerful and useful tools.

Authors' Note

Opinions expressed herein do not necessarily reflect those of the Institute for Education Sciences or related offices within.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: Research for the current study was partially supported by the Institute for Education Sciences Postdoctoral Fellowship Training Grant (R305B100016), Peabody Research Institute, Vanderbilt University.

References

Beath, K. (2015). Robust meta-analysis and meta-regression (Version 0.7-1) [Software]. Retrieved from project.org/web/packages/metaplus/metaplus.pdf

Borenstein, M., Hedges, L. V., Higgins, J. P. T., & Rothstein, H. R. (2010). A basic introduction to fixed-effect and random-effects models for meta-analysis. *Research Synthesis Methods*, *1*, 97–111.

Bulte, I. (2015). Single-case meta-analysis (Version 1.1.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/SCMA/SCMA.pdf

Burr, D. (2015). Bayesian multiple changepoint detection using metadata (Version 0.1-1) [Software]. Retrieved from http://cran.r-project.org/web/packages/bspmma/bspmma.pdf

Chen, H. (2015). Multivariate meta-analysis (Version 1.0) [Software]. Retrieved from http://cran.r-project.org/web/packages/mvtmeta/mvtmeta.pdf

Chen, D. G. D., & Peace, K. E. (2013). *Applied meta-analysis with R*. Boca Raton, FL: CRC Press.

Cheung, M. (2015). `metaSEM`: An R package for meta-analysis using structural equation modeling (Version 0.9-2) [Software]. Retrieved from https://courses.nus.edu.sg/course/psycwlm/Internet/metaSEM/

Crippa, A. (2015). Performing multivariate dose-response meta-analysis (Version 1.3.0) [Software]. Retrieved from http://cran.r-project.org/web/packages/dosresmeta/dosresmeta.pdf

Csardi, G. (2015). Download logs from the "RStudio" "CRAN" mirror (Version 2.1.0) [Software]. Retrieved from https://github.com/metacran/cranlogs

Debray, T. (2015). Diagnostic and prognostic meta-analysis (Version 0.1.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/metamisc/metamisc.pdf

De La Rue, L., Polanin, J. R., Espelage, D. L., & Pigott, T. D. (2016). A meta-analysis of school-based interventions aimed to prevent or reduce violence in teen dating relationships. *Review of Educational Research*. Advanced online publication. doi:10.3102/0034654316632061

Del Re, A. C. (2015a). Compute effect sizes (Version 0.2-4) [Software]. Retrieved from http://cran.r-project.org/web/packages/compute.es/compute.es.pdf

Del Re, A. C. (2015b). Graphical user interface for conducting meta-analyses in R (Version 0.0-2) [Software]. Retrieved from project.org/web/packages/RcmdrPlugin.MA/RcmdrPlugin.MA.pdf

Del Re, A. C., & Hoyt, W. T. (2015a). Meta-analysis with correlations (Version 1.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/MAc/MAc.pdf

Del Re, A. C., & Hoyt, W. T. (2015b). Meta-analysis with mean differences [Version 0.8.2] (Software). Retrieved from http://cran.r-project.org/web/packages/MAd/MAd.pdf

DerSimonian, R., & Laird, N. (1986). Meta-analysis in clinical trials. *Controlled Clinical Trials*, *7*, 177–188.

Dewey, M. (2015). Meta-analysis of significance values (Version 0.6) [Software]. Retrieved from http://cran.r-project.org/web/packages/metap/metap.pdf

Doebler, P. (2015). Meta-analysis of diagnostic accuracy (Version 0.5.7) [Software]. Retrieved from http://cran.r-project.org/web/packages/mada/mada.pdf

Fisher, Z., & Tipton, E. (2015). Robust variance meta-regression (Version 1.6) [Software]. Retrieved from http://cran.r-project.org/web/packages/robumeta/robumeta.pdf

Fletcher, T. D. (2015). Applied psychometric theory (Version 2.2) [Software]. Retrieved from http://cran.r-project.org/web/packages/psychometric/psychometric.pdf

Fox, J., Bouchet-Valat, M., Andronic, L., Ash, M., Boye, T., Calza, S., . . . Wright, K. (2016). R commander (Version 2.3-0) [Software]. Retrieved from https://cran.r-project.org/web/packages/Rcmdr/Rcmdr.pdf

Gasparrini, A. (2015). Multivariate and univariate meta-analysis and meta-regression (Version 0.4.7) [Software]. Retrieved from http://cran.r-project.org/web/packages/mvmeta/mvmeta.pdf

Ghosh, D., & Choi, H. (2015). Integration of microarray data for meta-analysis (Version 1.46.0) [Software]. Retrieved from http://www.bioconductor.org/packages/release/bioc/manuals/metaArray/man/metaArray.pdf

GitHub, Inc. (2016). Retrieved from https://github.com

Gordon, M., & Lumley, T. (2015). Advanced forest plot using "grid" graphics (Version 1.1) [Software]. Retrieved from https://cran.r-project.org/web/packages/forestplot/forestplot.pdf

Greenland, S., & Longnecker, M. P. (1992). Methods for trend estimation from summarized dose-response data, with applications to meta-analysis. *American Journal of Epidemiology*, *135*, 1301–1309.

Guolo, A., & Varin, C. (2015). Likelihood inference in meta-analysis and meta-regression models (Version 0.41.0) [Software]. Retrieved from http://cran.r-project.org/web/packages/metaLik/metaLik.pdf

Hamilton, W. K., Mizumoto, A., Aydin, B., & Coburn, K. (2015). Meta-analysis via shiny (Version 1.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/MAVIS/MAVIS.pdf

Hamling, J., Lee, P., Weitkunat, R., & Ambuhl, M. (2008). Facilitating meta-analyses by deriving relative effect and precision estimates for alternative comparisons from a set of estimates presented by exposure level or disease category. *Statistics in Medicine*, *27*, 954–970.

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis*. Orlando, FL: Academic.

Hedges, L. V., Tipton, E., & Johnson, M. C. (2010). Robust variance estimation in meta-regression with dependent effect size estimates. *Research Synthesis Methods*, *1*, 39–65.

Hedges, L. V., & Vevea, J. L. (1996). Estimating effect size under publication bias: Small sample properties and robustness of a random effects selection model. *Journal of Educational and Behavioral Statistics*, *21*, 299–333.

Huang, H. (2015). Meta-analysis for classification data and correction to imperfect reference (Version 0.1-1) [Software]. Retrieved from http://cran.r-project.org/web/packages/Metatron/Metatron.pdf

Huizenga, H. M., & Visser, I. (2015). Fit and test meta-regression models (Version 1.0-4) [Software]. Retrieved from http://cran.r-project.org/web/packages/metatest/metatest.pdf

Ihnatova, I. (2015). Meta-analysis of microarray (Version 2.2.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/MAMA/MAMA.pdf

Jackson, C. (2015). Ecological regression using aggregate and individual data (Version 0. 2) [Software]. Retrieved from https://cran.r-project.org/web/packages/ecoreg/ecoreg.pdf

Kang, D., & Tseng, G. (2015a). MetaPCA: Meta-analysis in the dimension reduction of genomic data (Version0.1.4) [Software]. Retrieved from http://cran.r-project.org/web/packages/MetaPCA/MetaPCA.pdf

Kang, D., & Tseng, G. (2015b). MetaQC: Objective quality control and inclusion/ exclusion criteria for genomic meta-analysis (Version 0.1.13) [Software]. Retrieved from http://cran.r-project.org/web/packages/MetaQC/MetaQC.pdf

Kovalchik, S. (2015). Tools for subgroup analysis with multiple trial data using aggregate statistics [Version 2.4] (Software). Retrieved from http://cran.r-project.org/web/packages/ipdmeta/ipdmeta.pdf

Kugler, K, Mueller, L., Ecker, S., & Graber, A. (2015). Meta-analytic data aggregation methods toolbox (Version 1.2.2) [Software]. Retrieved from https://r-forge.r-project .org/tracker/?atid=3068&group_id=774

Laliberté, E. (2015). Meta-analysis of correlation coefficients (Version 1.0-2) [Software]. Retrieved from http://cran.r-project.org/web/packages/metacor/metacor.pdf

Langan, D., Sutton, A., Higgins, J. P. T., & Gegory, W. (2015). Additional funnel plot augmentations (Version 1.3) [Software]. Retrieved from http://cran.r-project.org/web/packages/extfunnel/extfunnel.pdf

Lee, S. (2015). Meta-analysis for SNP-set (sequence) kernal association test (Version 0.40) [Software]. Retrieved from http://cran.r-project.org/web/packages/MetaSKAT/MetaSKAT.pdf

Lin, L., Zhang, J., & Chu, H. (2015). Patient-centered network meta-analysis (Version 2.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/pcnetmeta/pcnetmeta.pdf

Lumley, T. (2012). `rmeta`: Meta-analysis (Version 2.16) [Software]. Retrieved from http://cran.r-project.org/web/packages/rmeta/rmeta.pdf

Luo, S., Chen, Y., Su, X., & Chu, H. (2015). Multivariate meta-analysis (Version 2.2) [Software]. Retrieved from http://cran.r-project.org/web/packages/mmeta/mmeta.pdf

Lusa, L., Gentleman, R., & Ruschhaupt, M. (2015). Meta-analysis for high throughput experiments (Version 1.40.0) [Software]. Retrieved from http://www.bioconductor .org/packages/release/bioc/manuals/GeneMeta/man/GeneMeta.pdf

Marot, G. (2015). Meta-analysis for microarrarys (Version 3.1.2) [Software]. Retrieved from http://cran.r-project.org/web/packages/metaMA/metaMA.pdf

Marot, G., & Bruyère, R. (2015). Using metaMA for differential gene expression analysis from multiple studies. Retrieved from https://cran.r-project.org/web/packages/metaMA/vignettes/metaMA.pdf

Marot, G., Jaffrezic, F., & Rau, A. (2015). Meta-analysis of RNA-seq data (Version 1.0.2) [Software]. Retrieved from http://cran.r-project.org/web/packages/metaRNASeq/metaRNASeq.pdf

Möbius, T. W. D. (2015). Inference in meta-analysis and meta regression (Version 1.0) [Software]. Retrieved from http://cran.r-project.org/web/packages/metagen/metagen.pdf

Murrell, P., Scott, D., & Lu, E. (2014). Meta-analysis forest plots (Version 1.0) [Software]. Retrieved from https://github.com/pmur002/metaplot

Neupane, B., Richer, D., Bonner, A. J., Kibret, T., & Beyene, J. (2014) Network Meta-analysis using R: A review of currently retrieved automated packages. *PLoS One, 9*, *e115065*. doi:10.1371/journal.pone.0115065

Nikoloulopoulos, A. K. (2015). Copula mixed effect models for bivariate and trivariate meta-analysis of diagnostic test accuracy studies (Version 0.9) [Software]. Retrieved from https://cran.r-project.org/web/packages/CopulaREMADA/CopulaREMADA.pdf

Noory, Y. K. (2015). Sensitivity analysis of a meta-analysis with unpublished but registered analytical investigations (Version 1.2.1) [Software]. Retrieved from http://cran.r-project.org/web/packages/SAMURAI/SAMURAI.pdf

Pigott, T. D. (2012). *Advances in meta-analysis* (1st ed.). New York, NY: Springer.

Polanin, J. R., & Hennessy, E. A. (June, 2015). *A review of meta-analysis programs in R*. Paper presented at Use R! 2015, Aalborg, Denmark. Retrieved from http://rpubs.com/polanin/meta-analysis_rpackages_useR2015

R Core Team. (2015). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from http://www.R-project.org/

Revolution Analytics. (2016). *Applications of R*. Retrieved from http://www.revolutionanalytics.com/applications-r

Rexer Analytics. (2013). Retrieved from http://www.rexeranalytics.com/Data-Miner-Survey-2013-Intro.html

Rotondi, M. A. (2015). Sample size estimation functions for cluster randomized trials (Version 1.0) [Software]. Retrieved from https://cran.r-project.org/web/packages/CRTSize/CRTSize.pdf

Rücker, G. (2012). Network meta-analysis, electrical networks and graph theory. *Research Synthesis Methods*, *3*, 312–324. doi:10.1002/jrsm.1058

Rücker, G., Schwarzer, G., Krahn, U., & Konig, J. (2015). Network meta-analysis using frequentist methods (Version 0.7-0) [Software]. Retrieved from http://cran.r-project.org/web/packages/netmeta/netmeta.pdf

Rufibach, K. (2015). Estimation of weight functions in meta-analysis (Version 1.0.7) [Software]. Retrieved from http://cran.r-project.org/web/packages/selectMeta/selectMeta.pdf

Schiller, I., & Dendukuri, N. (2015). Meta-analysis of diagnostic test accuracy when reference test is imperfect (Version 2.1.8) [Software]. Retrieved from http://cran.r-project.org/web/packages/HSROC/HSROC.pdf

Schlattmann, P., Hoehne, J., & Verba, M. (2015). Finite mixture models and meta-analysis tools - based on CAMAN (Version 0.72) [Software]. Retrieved from http://cran.r-project.org/web/packages/CAMAN/CAMAN.pdf

Schwarzer, G. (2015). General package for meta-analysis (Version 4.2-0) [Software]. Retrieved from http://cran.r-project.org/web/packages/meta/meta.pdf

Schwarzer, G., Carpenter, J., & Rücker, G. (2015a). Advanced statistical methods to model and adjust for bias in meta-analysis (Version 0.2-0) [Software]. Retrieved from http://cran.r-project.org/web/packages/metasens/metasens.pdf

Schwarzer, G., Carpenter, J. R., & Rücker, G. (2015b). Heterogeneity and meta-regression. In G. Schwarzer, J. R. Carpenter, & G. Rücker *Meta-analysis with R* (pp. 85–104). Switzerland: Springer International Publishing.

Stack Exchange. (2016). Stack overflow. Retrieved from http://stackoverflow.com

Stevenson, M. (2015). Tools for the analysis of epidemiological data (Version 0.9-62) [Software]. Retrieved from http://cran.r-project.org/web/packages/epiR/epiR.pdf

Struchalin, M., & Aulchenko, Y. (2015). Meta-analysis of genome-wide SNP association results (Version 0.2-0)[Software]. Retrieved from http://cran.r-project.org/web/packages/MetABEL/MetABEL.pdf

Tanner-Smith, E. E., Tipton, E., & Polanin, J. R. (2016). Handling complex meta-analytic data structures using robust variance estimates: A tutorial in R. *Journal of Developmental and Life-Course Criminology*, *2*, 85–112. doi:10.1007/s40865-016-0026-5

Thornley, S. (2015). Performs simulation study to look for publication bias (Version 1.0) [Software]. Retrieved from http://cran.r-project.org/web/packages/PubBias/PubBias.pdf

van Valkenhoef, G., & Kuiper, J. (2014). Network meta-analysis using Bayesian Methods (Version 0.6-1) [Software]. Retrieved from http://cran.r-project.org/web/packages/gemtc/gemtc.pdf

Verde, P. E., & Sykosch, A. (2015). Bayesian meta-analysis of diagnostic test data [Version 1.1-1] (Software). Retrieved from: http://cran.r-project.org/web/packages/bamdit/bamdit.pdf

Veroniki, A. A., Jackson, D., Viechtbauer, W., Bender, R., Bowden, J., Knapp, G., … Salanti, G. (2016). Methods to estimate the between-study variance and its uncertainty in meta-analysis. *Research Synthesis Methods*, *7*, 55–79. doi:10.1002/jrsm.1164

Vevea, J. L., & Coburn, K. (2015). The Vevea and Hedges weight-function model for publication bias (Version 1.0) [Software]. Retrieved from https://vevealab.shinyapps.io/WeightFunctionModel/

Vevea, J. L., & Hedges, L. V. (1995). A general linear model for estimating effect size in the presence of publication bias. *Psychometrika*, *60*, 419–435.

Viechtbauer, W. (2015). Metafor: Meta-analysis package for R (Version 1.9-5) [Software]. Retrieved from http://cran.r-project.org/web/packages/metafor/metafor.pdf

Voorman, A., Brody, J., Chen, H., & Lumley, T. (2015). Meta-analysis of region-based tests of rare DNA variants (Version 1.5) [Software]. Retrieved from http://cran.r-project.org/web/packages/seqMeta/seqMeta.pdf

Voorman, A., Brody, J., & Lumley, T. (2015). Efficient meta-analysis for the SKAT test [Version 1.4.3] (Software). Retrieved from http://cran.r-project.org/web/packages/skatMeta/skatMeta.pdf

Vuckovic, D. (2015). Meta-analysis of multivariate genome wide association studies (Version 0.1) [Software]. Retrieved from https://cran.r-project.org/web/packages/MultiMeta/MultiMeta.pdf

Wang, X., Li, J., & Tseng, G. C. (2015). MetaDE: Microarray meta-analysis for differentially expressed gene detection (Version 1.0.5) [Software]. Retrieved from http://cran.r-project.org/web/packages/MetaDE/MetaDE.pdf

Wickham, H. (2009). *ggplot2: Elegant graphics for data analysis*. New York, NY: Springer Science & Business Media.

Wickham, H., & Maeland, B. (2016). *Crantastic*! Retrieved from http://crantastic.org

Williams, R. T. (2012). *Using robust standard errors to combine multiple estimates with meta-analysis*. (Unpublished Doctoral Dissertation). Loyola University Chicago, Chicago, IL.

Wilson, S. J., Polanin, J. R., & Lipsey, M. W. (2016). Meta-analyzing a complex correlational dataset. *Research Synthesis Methods*, *7*, 121–139. doi:10.1002/jrsm.1199

Winker, T. (2015). Evaluation of stratified genome-wide association meta-analysis (Version 8.6) [Software]. Retrieved from http://cran.r-project.org/web/packages/EasyStrata/EasyStrata.pdf

Yang, G., Shi, P., & Xie, M. (2016). Meta-analysis via a unified framework under confidence distribution (Version 2.2-3) [Software]. Retrieved from https://cran.r-project.org/web/packages/gmeta/gmeta.pdf

Yu, Y., & Tian, L. (2015). Exact fixed effect meta-analysis (Version 1.0-2) [Software]. Retrieved from http://cran.r-project.org/web/packages/exactmeta/exactmeta.pdf

Zhao, J. H. (2015). Genetic analysis package (Version 1.1-16) [Software]. Retrieved from http://cran.r-project.org/web/packages/gap/gap.pdf

Authors

JOSHUA R. POLANIN, PhD, is a senior research scientist at Development Services Group, Inc. (a social science research firm that specializes in training and technical assistance, impact evaluations, and large-scale systematic reviews); 7315 Wisconsin Ave., Suite 840, Bethesda, MD 20814; email: jpolanin@dsgonline.com. His areas of expertise are in adolescent health and well-being, school violence, and applied statistics, especially systematic review and meta-analysis. He serves as the senior methodologist for the Substance Abuse and Mental Health Services Administration's National Registry of Evidence-based Programs and Practices (NREPP), is the principal investigator of a recently awarded National Institutes of Justice grant to conduct a large-scale systematic review and meta-analysis on the long-term consequences of school violence, and is secretary and co-program chair of the Systematic Review and Meta-Analysis Special Interest Group within the American Education Research Association.

EMILY A. HENNESSY, MPhil, is a doctoral candidate at the Peabody College of Human and Organizational Development at Vanderbilt University, Nashville, TN, and an expert in adolescent health promotion; email: emily.a.hennessy@vanderbilt.edu. Her dissertation focuses on factors affecting adolescent recovery from substance use disorders. She is also a systematic review specialist for the International Initiative for Impact Evaluation and a graduate student reviewer for the Methods Coordinating Group of the Campbell Collaboration.

EMILY E. TANNER-SMITH, PhD, is an associate research professor at the Peabody Research Institute and Department of Human and Organizational Development at Vanderbilt University, Nashville, TN, and is the codirector of the Meta-Analysis Center at the Peabody Research Institute; email: e.tanner-smith@vanderbilt.edu. Her areas of expertise are in substance use and addiction, adolescent behavior and development, and applied research methods.

Manuscript received April 27, 2016
Revision received August 13, 2016
Accepted September 10, 2016