

A contrast of meta and metafor packages for meta-analyses in R

Christopher J. Lortie^{1,2} and Alex Filazzola³

1. Department of Biology, York University
Toronto, ON, Canada. M3J1P3.
lortie@yorku.ca
PH: 416.736.2100 x20588
2. The National Center for Ecological Analysis and Synthesis,
UCSB, California. USA.
3. Biological Sciences, University of Alberta,
Edmonton, AB, Canada,

Abstract

There is extensive choice in R to support meta-analyses. Two common packages used in the sciences include meta and metafor. A brief contrast of their strengths is described here for the synthesis scientist. Meta is a direct, intuitive choice for rapid implementation of general meta-analytical statistics. Metafor is a comprehensive package best suited for relatively more complex models. Both packages provide estimates of heterogeneity, excellent visualization tools, and functions to explore publication bias. Preference and specific needs will facilitate choice between these two options. Nonetheless, metafor has a steeper learning curve but greater rewards.

Keywords

contrasts, computational tools, Meta, meta-analyses, Metafor, methods, R, review

Introduction

Meta-analyses are common and powerful synthesis tools in science. Typically in the natural sciences, meta-analyses are used as a mechanism to describe and aggregate quantitative evidence from a set of peer-reviewed, primary research publications (Lortie and Bonte 2016; Nakagawa et al. 2017). A meta-analysis in the natural sciences, i.e. outside the health sciences, is comprised of a formalized systematic review and analysis of effect sizes and is termed a meta-analysis when statistics examining intervention efficacy are included (Lortie 2014). In other fields, the terms systematic review and meta-analysis are used more interchangeably, and meta-statistics are often done on compiled randomized controlled trials or other relatively large datasets in addition to data derived from peer-reviewed publications. The derived data in the natural sciences routinely extracts only the mean values or single point measures from publications (Stewart and Schmid 2015). The synthesis statistics were commonly done using MetaWin in some fields (Rosenberg, Adams, and Gurevitch 2000) or other GUI-based applications for a number of years. More recently however, statistics in the fields of ecology and evolution for instance have increasingly moved to the programming language R (Lai et al. 2019), and synthesis statistics are no exception. At least two R packages have risen to prominence for general meta-analytical statistics in the natural sciences - namely *meta* (Schwarzer 2019) and *metafor* (Viechtbauer 2017). Given that meta-analyses are also increasingly published in these same fields (Cadotte, Mehrkens, and Menge 2012; Lortie and Bonte 2016), a brief comment on the ecosystem of analytical choices that R provides is beneficial and timely. We need synthesis to inform evidence-based decisioning, and meta-analyses can be the primary tool if aggregated primary datasets are unavailable. Furthermore even with primary data in hand, data reduction to effect sizes within primary and synthesis studies is a mechanism to illustrate differences and strength of effects. These approaches provide the capacity for higher-order analyses and reuse (Gerstner et al. 2017) suggesting that familiarity with effect sizes is both germane and practical.

The R ecosystem for meta-analyses

Like many fundamental challenges in science, the R developer community provides potential solution sets distributed across multiple packages for synthesis. Broadly speaking, alternative packages in R sometimes examine an issue from different perspectives and provide unique functions. In other instances, packages can be very similar or analogs in terms of functionality and use conceptually aligned functions that differ only in nomenclature or arguments (Lortie et al. 2020). Scientific synthesists that choose to do a meta-analysis in R have options. A total of 63 packages associated with various aspects of conducting a meta-analysis have been identified in a comprehensive review and typology of options (Polanin, Hennessy, and Tanner-Smith 2016). Both *meta* (Schwarzer 2019) and *metafor* (Viechtbauer 2017) are amongst 11 generic packages identified (Polanin, Hennessy, and Tanner-Smith 2016). These two packages are analogs but with different inherent workflows. There is also *rmeta* for simple fixed and random effects meta-analyses (Lumley 2018), *mada* for diagnostics (Doebler 2017), *netmeta* for frequentist network meta statistics (Rucker et al. 2019), and *mvmeta* for multivariate derived data aggregations (Gasparrini 2018) to name a few options. The latter three packages listed have distinct and specific niches for analysis whilst *meta* and *metafor* overlap considerably. Consequently, a brief contrast in facilitating choice between these two packages for the general analyst is provided here.

Contrast of meta versus metafor

Meta is a well-maintained, recently updated CRAN R package (Version 4.11-0 updated on Feb 20, 2020) with 31 unique functions, 7 sample datasets (Appendix A), and a reference manual. There is also an exceptional textbook devoted to meta-analysis in R that focuses primarily on this package (Schwarzer, Carpenter, and Rücker 2015). It is highly capable of resolving most general meta-analytical challenges that an analyst will face including the capacity to include Empirical Bayes estimators as arguments in some functions, predictive meta-statistics, interaction terms, meta-regression, and modifiers. Note that for some of these methods, the *rma.uni* function is sourced internally from the package *metafor*. This is intriguing but mostly opaque and inconsequential to the user if she prefers the structure of the arguments within functions, the semantics, or the workflow of the *meta* package. The primary strengths include its direct and straightforward implementation with minimal (source) lines of code to do an analysis. Provided one has secured the derived data from the studies and organized into a dataframe with vectors as each key argument within the main *meta*-model

fitting functions, statistics are simple. The type of response variable such as mean, continuous, or rate is matched to a specific function call such as `metamean`, `metacont`, or `metarate`. This is semantically intuitive and encourages good thinking before statistics because it engenders consideration of the data. The effect size calculation is included in this main function and defaults return the most prevalent effect size measure typically associated with those data, but it can also be specified as an argument. The primary workflow is thus a single step if the user elects to rely on the internal calculations provided in this package. Exploration of the model is well articulated with funnel, radial, and forest plots. Z-scores, significance tests, and heterogeneity statistics are printed in the model summary. Publication bias is also provided as a more in-depth function entitled `metabias` within this package. There are two standout functions in this package. The first is a function entitled `metagen`, and it is a backup, multipurpose tool so to speak that fits a generic inverse variance meta-analysis. This is a handy tool for user-calculated effect size measures or for exploration of statistical trends with reduced data assumptions. In some fields, there are specific effect size estimates that this function provides a robust, easy-to-fit capacity for statistics. The second standout function is `bubble.metareg` for a quick, visual exploration of the outcome of a meta-regression. It is useful in contemporary data science to use visualization as a means to understand data (Lortie 2017), but statistical packages do not always provide the means to easily iterate between statistics or model fitting and visualization. In summary, excepting unique data or statistical issues, this package is directly implemented and effective.

Metafor is a more comprehensive package in many respects. This package includes 74 functions, 35 datasets (Appendix A), a vignette (Viechtbauer 2010), flowchart as secondary vignette (<https://cran.r-project.org/web/packages/metafor/vignettes/diagram.pdf>), and website (<http://www.metafor-project.org/doku.php>). The package was last updated on March 19, 2020 (Version 2.4-0). The text ‘Meta-analysis with R’ also describes implementation of this package (Schwarzer, Carpenter, and Rücker 2015) but to a lesser extent than meta. The depth of the package metafor provides greater capacities relative to the meta package but does come at the expense of a steeper initial learning curve. Completing a meta-analysis using this package requires an additional step, i.e. effect sizes must be calculated a priori, not within the model fitting process. This is facilitated with the standalone function `escalc`, and it can return a wide range of effect sizes measures. Thus, the two-step process begins with firstly compiling and aggregating the derived dataframe to an effect size table then secondly fitting a model. The data structure is also a bit more rigid for the model fitting, and the nomenclature for this subset of functions is written to parallel more traditional general linear model fitting from conventional statistics. This is both a strength and limitation because one must plan the model to fit in advance and learn the function and arguments, but it is also an advantage as well because model specification uses the familiar notation of tilde. Model fitting is based on the type of model in the call such as a random or fixed effects and not on the type of the response data as in meta package. Here, it is more akin to conventional general linear model fitting for those familiar with these functions in R. If the model is more complex with moderators, then this can be directly included in the model fit here via a `mods` argument whereas in the meta package the model is updated with moderators in a subsequent step. This suggests that if moderators or covariates in the main model are likely relevant to the analyses, then metafor is a strong starting point. The model summary also prints Z-scores, significance tests, and two sets of heterogeneity estimates. Forest and radial plots are also provided as additional functions. Publication bias statistical estimator functions include `trimfill` and `ranktest`. Standout elements of this package include GOSH plots that provide a graphical display of study heterogeneity (Olkin, Dahabreh, and Trikalinos 2012) and the enhanced model fitting capacities such as the function `fitstats` that provides log-likelihood estimates and AIC or BIC scores on meta-analysis objects. This package requires a focus on model fitting, and while there is additional effort in specifying the data at the onset of the workflow, the rewards in subsequent tools to handle models are significant.

Models are powerful and typically necessary tools to best understand datasets. Conservatively, a model is a mechanism to explore uncertainty and assign weight to an observed pattern in the data [Glad and Hjort (2016)]. It has been proposed that all models are wrong but that some are useful (Stouffer 2019).

Conclusions

Statistics are sometimes about preferences and thinking styles (Hector 2017), and scientific synthesis is both an art and a science (Lortie and Bonte 2016). Trade-offs are also common in adopting one ecosystem,

analysis tool, or specific package for data wrangling and analyses. If more rapid, less specified, general meta-analyses are the goal – the package `meta` is a direct means to an end. Moderators are added post hoc in additional, update model steps, but the first model fit is a single, intuitive process. Meta-regression is viable and interaction terms can be included. The generic meta-analysis function is a superb tool. `Metafor` requires the effect size compilation a priori and is thus a bit more coding to prepare for the meta-model. However, deeper and more complex model fits are inherent in the semantics of these functions. If the synthesist does have not effect size measure in hand or wishes to calculate effect sizes measures but not for meta-models, the `escalc` function is invaluable in this package. In summary, both packages provide the capacity for basic and advanced meta-analyses but more advanced modelling is likely worth the commitment to `metafor`.

Table 1.

A contrast of meta and metafor using the critical principles developed in ‘A checklist for choosing between R packages in ecology and evolution’ (Lortie et al. 2020).

item	criteria	meta	metafor
1	maturity	fourth major version	third major version
2	active development	Schwarzer maintains website and GitHub repository	Viechtbauer maintains website and GitHub repository
3	recently updated	Feb 20, 2020	March 19, 2020
4	documentation available	reference manual	reference manual and two vignettes
5	published in similar projects	widespread use in ecology, evolution, and to a lesser extent the social sciences	widespread use in ecology, evolution, and the social sciences
6	license	GPL-2	GPL-2
7	semantics intuitive	functions to fit meta-analyses corresponds to data type intuitive	fitting meta-analyses logic is similar to GLM fitting in R
8	functions that get the job done	most major data types have a specific function and there is also a generalized metagen function	requires clear specification in arguments but rma function is a very general yet flexible tool
9	arguments to support your needs	there is an sm argument for specification of summary measures within meta-analysis functions	extensive breadth in capacity to specify model fitting
10	dependencies reasonable and reported	depends on R, and imports include grid, metafor, lme4, and CompQuadForm listed	depends on R, and imports include stats, utils, graphics, grDevices, and nlme

Literature cited

- Cadotte, M.W., L.R. Mehrkens, and D.N.L. Menge. 2012. “Gauging the Impact of Meta-Analysis on Ecology.” Journal Article. *Evolutionary Ecology* 26: 1153–67.
- Doebler, P. 2017. “Mada.” Journal Article. *CRAN*, Version 0.5.8.
- Gasparrini, A. 2018. “Mvmeta.” Journal Article. *CRAN*, Version 0.4.11.
- Gerstner, Katharina, David Moreno-Mateos, Jessica Gurevitch, Michael Beckmann, Stephan Kambach, Holly P. Jones, and Ralf Seppelt. 2017. “Will Your Paper Be Used in a Meta-Analysis? Make the Reach of Your Research Broader and Longer Lasting.” Journal Article. *Methods in Ecology and Evolution* 8 (6): 777–84. <https://doi.org/10.1111/2041-210X.12758>.
- Glad, Ingrid, and Nils Lid Hjort. 2016. “Model Uncertainty First, Not Afterwards.” Journal Article. *Statist. Sci.* 31 (4): 490–94. <https://doi.org/10.1214/16-STS559>.
- Hector, A. 2017. *The New Statistics with R*. Book. Oxford: Oxford University Press.
- Lai, Jiangshan, Christopher J. Lortie, Robert A. Muenchen, Jian Yang, and Keping Ma. 2019. “Evaluating the Popularity of R in Ecology.” Journal Article. *Ecosphere* 10 (1): e02567. <https://doi.org/10.1002/ecs2.2567>.
- Lortie, Christopher. 2017. “R for Data Science.” Journal Article. *Journal of Statistical Software* 77: 1–3. <https://doi.org/10.18637/jss.v077.b01>.
- Lortie, Christopher J., and Dries Bonte. 2016. “Zen and the Art of Ecological Synthesis.” Journal Article. *Oikos* 125 (3): 285–87. <https://doi.org/10.1111/oik.03161>.
- Lortie, Christopher J., Jenna Braun, Alessandro Filazzola, and Florencia Miguel. 2020. “A Checklist for Choosing Between R Packages in Ecology and Evolution.” Journal Article. *Ecology and Evolution*. <https://doi.org/10.1002/ece3.5970>.
- Lortie, C. J. 2014. “Formalized Synthesis Opportunities for Ecology: Systematic Reviews and Meta-Analyses.” Journal Article. *Oikos* 123: 897–902.
- Lumley, T. 2018. “Rmeta.” Journal Article. *CRAN*, Version 3.0.
- Nakagawa, Shinichi, Daniel W. A. Noble, Alistair M. Senior, and Malgorzata Lagisz. 2017. “Meta-Evaluation of Meta-Analysis: Ten Appraisal Questions for Biologists.” Journal Article. *BMC Biology* 15 (1): 18. <https://doi.org/10.1186/s12915-017-0357-7>.
- Olkin, Ingram, Issa J. Dahabreh, and Thomas A. Trikalinos. 2012. “GOSH – a Graphical Display of Study Heterogeneity.” Journal Article. *Research Synthesis Methods* 3 (3): 214–23. <https://doi.org/10.1002/jrsm.1053>.
- Polanin, Joshua R., Emily A. Hennessy, and Emily E. Tanner-Smith. 2016. “A Review of Meta-Analysis Packages in R.” Journal Article. *Journal of Educational and Behavioral Statistics* 42 (2): 206–42. <https://doi.org/10.3102/1076998616674315>.
- Rosenberg, Michael, Dean Adams, and Jessica Gurevitch. 2000. *MetaWin: Statistical Software for Meta-Analysis. Version 2.0*. Book.
- Rucker, G., U. Krahn, J. König, O. Efthimious, and G. Schwarzer. 2019. “Netmeta.” Journal Article. *CRAN* 1.0.1: Version 1.0.1.
- Schwarzer, G. 2019. “Meta.” Journal Article. *CRAN*, Version 4.94.
- Schwarzer, G., J.R. Carpenter, and G. Rücker. 2015. *Meta-Analysis with R*. Book. Switzerland: Springer.
- Stewart, G. B., and C. H. Schmid. 2015. “Lessons from Meta-Analysis in Ecology and Evolution: The Need for Trans-Disciplinary Evidence Synthesis Methodologies.” Journal Article. *Research Synthesis Methods* 6: 109–10. <https://doi.org/10.1002/jrsm.1152>.

- Stouffer, Daniel B. 2019. “All Ecological Models Are Wrong, but Some Are Useful.” Journal Article. *Journal of Animal Ecology* 88 (2): 192–95. <https://doi.org/10.1111/1365-2656.12949>.
- Viechtbauer, W. 2010. “Conducting Meta-Analyses in R with the Metafor Package.” Journal Article. *J Stat Software* 36. <https://doi.org/10.18637/jss.v036.i03>.
- . 2017. “Metafor.” Journal Article. *CRAN*, Version 2.0.