# An Overview of Functions in the *metafor* Package

last updated: Oct 16 2022
(not all functions documented)

given the required data (e.g., means, SDs, and group sizes; counts for 2x2 tables; correlations and sample sizes), calculate the desired effect size or outcome measure for the meta-analysis (e.g., standardized mean differences, log odds or risk ratios, risk differences, r-to-z transformed correlations, …) and the corresponding sampling variances (or an entire variance-covariance matrix for dependent estimates)

**read.table()**
**read.csv()**
**read.delim()**

functions in the 'util' package to:
- read in data from ASCII file
- see also 'foreign', 'readxl', and 'haven' packages for reading in other data formats

**escalc()**
**vcalc()**
**rcalc()**

- yi = observed outcomes or effect size estimates
- vi = corresponding sampling variances (or 'V' for an entire var-cov matrix)

**rma.uni()**
**rma.mh()**
**rma.peto()**
**rma.glmm()**
**rma.mv()**

- rma.uni() = equal/fixed- and random/mixed-effects models ("inverse-variance" method; normal-normal models)
- rma.mh() = Mantel-Haenszel method
- rma.peto() = Peto's method (equal/fixed-effects model)
- rma.glmm() = equal/fixed- and random/mixed-effects models (binomial-normal and Poisson-normal models)
- rma.mv() = equal/fixed- and random/mixed-effects multivariate/multilevel models (normal-normal models)

note: rma.uni() takes either 'yi' and 'vi' as input or one can supply the required data to calculate the desired effect size or outcome measure for the meta-analysis directly; rma.mh(), rma.peto(), and rma.glmm() require that the raw counts are supplied; rma.mv() takes 'yi' and 'V' as input (V is the variance-covariance matrix of the sampling errors)

**print()**
**summary()**
**aggregate()**

*print functions*

**print()**
**summary()**

note: class of fitted model object is the same as the function name; so print() for an object of class 'rma.uni' actually calls print.rma.uni() and so on

*fitted and predicted values*

**fitted()**
**predict()**
**blup()**
**ranef()**
**cumul()**

note: blup() only for 'rma.uni' objects; ranef() only for 'rma.uni' and 'rma.mv' objects; cumul() not for 'rma.mv' or 'rma.glmm' objects

*residuals and influential case diagnostics*

**residuals()**
**rstandard()**
**rstudent()**
**hatvalues()**
**weights()**
**influence()**
**leave1out()**

note: all functions implemented for 'rma.uni' objects; coverage of functions for other objects varies (see docs)

*funnel plot asymmetry / publication bias*

**ranktest()**
**regtest()**
**trimfill()**
**hc()**
**tes()**
**selmodel()**

note: regtest() not for 'rma.glmm' or 'rma.mv' objects; trimfill(), hc(), tes(), selmodel() only for 'rma.uni' objects

*confidence intervals and inference*

**confint()**
**anova()**
**permutest()**
**robust()**
**vif()**

note: confint() not for 'rma.glmm' objects; anova() and robust() only for 'rma.uni' and 'rma.mv' objects; permutest() only for 'rma.uni' objects

*plotting functions*

**forest()**
**funnel()**
**labbe()**
**radial()**
**qqnorm()**
**baujat()**
**gosh()**
**regplot()**
**plot()**

note: forest() and funnel() also take 'yi' and 'vi' as input; qqnorm(), baujat(), gosh() and plot() not for 'rma.glmm' or 'rma.mv' objects

*various extractor functions*

**logLik()**
**deviance()**
**fitstats()**
**AIC(), BIC()**
**coef()**
**vcov()**

note: coef() also for 'permutest.rma.uni' and 'summary.rma' objects