Package ‘altmeta’

July 24, 2020

Type Package

Title Alternative Meta-Analysis Methods

Version 3.1

Date 2020-07-23

Maintainer Lifeng Lin <linl@stat.fsu.edu>

Depends R (>= 3.5.0)

Imports rjags (>= 4-6), coda, graphics, grDevices, lme4, Matrix, metafor, methods, stats, utils

SystemRequirements JAGS 4.x.y (http://mcmc-jags.sourceforge.net)

Description Provides alternative statistical methods for meta-analysis, including:
- new heterogeneity tests and measures that are robust to outliers
(Lin et al., 2017 <doi:10.1111/biom.12543>);
- measures, tests, and visualization tools for publication bias or small-study effects
(Lin and Chu, 2018 <doi:10.1111/biom.12817>; Lin, 2019 <doi:10.1002/jrsm.1340>;
Lin, 2020 <doi:10.1177/0962280220910172>; Shi et al., 2020 <doi:10.1002/jrsm.1415>);
- meta-analysis methods for synthesizing proportions
(Lin and Chu, 2020 <doi:10.1097/ede.0000000000001232>);
- models for multivariate meta-analysis
(Lin and Chu, 2018 <doi:10.1002/jrsm.1293>).

NeedsCompilation no

Author Lifeng Lin [aut, cre] (<https://orcid.org/0000-0002-3562-9816>),
Linyu Shi [ctb],
Haitao Chu [aut] (<https://orcid.org/0000-0003-0932-598X>)

Repository CRAN

Date/Publication 2020-07-24 12:50:02 UTC

R topics documented:

`dat.aex` *A Meta-Analysis for Evaluating the Effect of Aerobic Exercise on Visceral Adipose Tissue Content/Volume*

Description

This meta-analysis serves as an example to illustrate function usage in the package **altmeta**.

Usage

```
data("dat.aex")
```

Format

A data frame containing 29 studies with the observed effect sizes and their within-study variances.

`y` the observed effect size for each collected study in the meta-analysis.

`s2` the within-study variance for each study.

Source

Ismail I, Keating SE, Baker MK, Johnson NA (2012). "A systematic review and meta-analysis of the effect of aerobic vs. resistance exercise training on visceral fat." *Obesity Reviews*, **13**(1), 68–91. <doi: 10.1111/j.1467789X.2011.00931.x>

`dat.annane` *A Meta-Analysis for Comparing the Effect of Steroids vs. Control in the Length of Intensive Care Unit (ICU) Stay*

Description

This dataset serves as an example of meta-analysis of mean differences.

Usage

```
data("dat.annane")
```

Format

A data frame with 12 studies with the following 5 variables within each study.

`y` point esimates of mean differences.

`s2` sample variances of mean differences.

`n1` sample sizes in treatment group 1 (steroids).

`n2` sample sizes in treatment group 2 (control).

`n` total sample sizes.

Source

Annane D, Bellissant E, Bollaert PE, Briegel J, Keh D, Kupfer Y (2015). "Corticosteroids for treating sepsis." *Cochrane Database of Systematic Reviews*, **12**, Art. No.: CD002243. <doi: 10.1002/14651858.CD002243.pub3>

`dat.barlow` *A Meta-Analysis on the Effect of Parent Training Programs vs. Control for Improving Parental Psychosocial Health Within 4 Weeks After Intervention*

Description

This dataset serves as an example of meta-analysis of standardized mean differences.

Usage

```
data("dat.barlow")
```

Format

A data frame with 26 studies with the following 5 variables within each study.

- `y` point esimates of standardized mean differences.
- `s2` sample variances of standardized mean differences.
- `n1` sample sizes in treatment group 1 (parent training programs).
- `n2` sample sizes in treatment group 2 (control).
- `n` total sample sizes.

Source

Barlow J, Smailagic N, Huband N, Roloff V, Bennett C (2014). "Group-based parent training programmes for improving parental psychosocial health." *Cochrane Database of Systematic Reviews*, **5**, Art. No.: CD002020. <doi: 10.1002/14651858.CD002020.pub4>

`dat.beck17` *A Meta-Analysis of Prevalence of Depression or Depressive Symptoms Among Medical Students*

Description

This dataset serves as an example of meta-analysis of proportions.

Usage

```
data("dat.beck17")
```

Format

A data frame with 6 studies with the following 2 variables within each study.

- `e` event counts of samples with depression or depressive symptoms.
- `n` sample sizes.

Details

The original article by Rotenstein et al. (2016) stratified all extracted studies based on various screening instruments and cutoff scores. This dataset focuses on the meta-analysis of 6 studies with Beck Depression Inventory Score $\geq$17.

Source

Rotenstein LS, Ramos MA, Torre M, Segal JB, Peluso MJ, Guille C, Sen S, Mata DA (2016). "Prevalence of depression, depressive symptoms, and suicidal ideation among medical students: a systematic review and meta-analysis." *JAMA*, **316**(21), 2214–2236. <doi: 10.1001/jama.2016.17324>

`dat.butters` *A Meta-Analysis on the Overall Response of the Addition of Drugs to a Chemotherapy Regimen for Metastatic Breast Cancer*

Description

This dataset serves as an example of meta-analysis of (log) odds ratios.

Usage

```
data("dat.butters")
```

Format

A data frame with 16 studies with the following 7 variables within each study.

- `y` point esimates of log odds ratios.
- `s2` sample variances of log odds ratios.
- `n1` sample sizes in treatment group 1 (addition of drug).
- `n2` sample sizes in treatment group 2 (control).
- `r1` event counts in treatment group 1.
- `r2` event counts in treatment group 2.
- `n` total sample sizes.

Source

Butters DJ, Ghersi D, Wilcken N, Kirk SJ, Mallon PT (2010). "Addition of drug/s to a chemotherapy regimen for metastatic breast cancer." *Cochrane Database of Systematic Reviews*, **11**, Art. No.: CD003368. <doi: 10.1002/14651858.CD003368.pub3>

`dat.chor` *A Meta-Analysis of Proportions on Chorioamnionitis*

Description

This dataset serves as an example of meta-analysis of proportions.

Usage

```
data("dat.chor")
```

Format

A data frame with 21 studies with the following 2 variables within each study.

`e` event counts of horioamnionitis.

`n` sample sizes.

Source

Woodd SL, Montoya A, Barreix M, Pi L, Calvert C, Rehman AM, Chou D, Campbell OMR (2019). "Incidence of maternal peripartum infection: a systematic review and meta-analysis." *PLOS Medicine*, **16**(12), e1002984. <doi: 10.1371/journal.pmed.1002984>

`dat.ducharme` *A Meta-Analysis on the Effect of Long-Acting Inhaled Beta2-Agonists vs. Control for Chronic Asthma*

Description

This meta-analysis serves as an example of meta-analysis with binary outcomes.

Usage

```
data("dat.ducharme")
```

Format

A data frame containing 33 studies with the following 4 variables within each study.

`n00` counts of non-events in treatment group 0 (placebo).

`n01` counts of events in treatment group 0 (placebo).

`n10` counts of non-events in treatment group 1 (beta2-agonists).

`n11` counts of events in treatment group 1 (beta2-agonists).

Note

The original review collected 35 studies; two double-zero-counts studies are excluded from this dataset because their odds ratios are not estimable.

Source

Ducharme FM, Ni Chroinin M, Greenstone I, Lasserson TJ (2010). "Addition of long-acting beta2-agonists to inhaled corticosteroids versus same dose inhaled corticosteroids for chronic asthma in adults and children." *Cochrane Database of Systematic Reviews*, **5**, Art. No.: CD005535. <doi: 10.1002/14651858.CD005535.pub2>

`dat.fib` *A Multivariate Meta-Analysis by the Fibrinogen Studies Collaboration*

Description

This multivariate meta-analysis serves as an example to illustrate function usage in the package **altmeta**. It consists of 31 studies with 4 outcomes.

Usage

```
data("dat.fib")
```

Format

A list containing three elements, `y`, `S`, and `sd`.

`y` a 31 x 4 numeric matrix containing the observed effect sizes; the rows represent studies and the columns represent outcomes.

`S` a list containing 31 elements; each element is within-study covariance matrix of the corresponding study.

`sd` a 31 x 4 numeric matrix containing the within-study standard deviations; the rows represent studies and the columns represent outcomes.

Source

Fibrinogen Studies Collaboration (2004). "Collaborative meta-analysis of prospective studies of plasma fibrinogen and cardiovascular disease." *European Journal of Cardiovascular Prevention and Rehabilitation*, **11**(1), 9–17. <doi: 10.1097/01.hjr.0000114968.39211.01>

Fibrinogen Studies Collaboration (2005). "Plasma fibrinogen level and the risk of major cardiovascular diseases and nonvascular mortality: an individual participant meta-analysis." *JAMA*, **294**(14), 1799–1809. <doi: 10.1001/jama.294.14.1799>

`dat.ha` *A Meta-Analysis on the Effect of Placebo Interventions for All Clinical Conditions Regarding Patient-Reported Outcomes*

Description

This meta-analysis serves as an example to illustrate function usage in the package **altmeta**.

Usage

```
data("dat.ha")
```

Format

A data frame containing 109 studies with the observed effect sizes and their within-study variances.

`y` the observed effect size for each collected study in the meta-analysis.

`s2` the within-study variance for each study.

Source

Hrobjartsson A, Gotzsche PC (2010). "Placebo interventions for all clinical conditions." *Cochrane Database of Systematic Reviews*, **1**. Art. No.: CD003974. <doi: 10.1002/14651858.CD003974.pub3>

`dat.henry` *A Meta-Analysis for Evaluating the Effect of Tranexamic Acid on Perioperative Allogeneic Blood Transfusion*

Description

This meta-analysis serves as an example of meta-analysis with binary outcomes.

Usage

```
data("dat.henry")
```

Format

A data frame containing 26 studies with the following 4 variables within each study.

`n00` counts of non-events in treatment group 0 (placebo).

`n01` counts of events in treatment group 0 (placebo).

`n10` counts of non-events in treatment group 1 (tranexamic acid).

`n11` counts of events in treatment group 1 (tranexamic acid).

Note

The original review collected 27 studies; one double-zero-counts study is excluded from this dataset because its odds ratio is not estimable.

Source

Henry DA, Carless PA, Moxey AJ, O'Connell, Stokes BJ, Fergusson DA, Ker K (2011). "Anti-fibrinolytic use for minimising perioperative allogeneic blood transfusion." *Cochrane Database of Systematic Reviews*, **1**, Art. No.: CD001886. <doi: 10.1002/14651858.CD001886.pub3>

`dat.hipfrac` *A Meta-Analysis on the Magnitude and Duration of Excess Mortality After Hip Fracture Among Older Men*

Description

This meta-analysis serves as an example to illustrate function usage in the package **altmeta**.

Usage

```
data("dat.hipfrac")
```

Format

A data frame containing 17 studies with the observed effect sizes and their within-study variances.

`y` the observed effect size for each collected study in the meta-analysis.

`s2` the within-study variance for each study.

Source

Haentjens P, Magaziner J, Colon-Emeric CS, Vanderschueren D, Milisen K, Velkeniers B, Boonen S (2010). "Meta-analysis: excess mortality after hip fracture among older women and men". *Annals of Internal Medicine*, **152**(6), 380–390. <doi: 10.7326/000348191526201003160000008>

dat.kaner *A Meta-Analysis on the Effect of Brief Alcohol Interventions vs. Control in Primary Care Populations*

Description

This dataset serves as an example of meta-analysis of risk differences.

Usage

```
data("dat.kaner")
```

Format

A data frame with 13 studies with the following 7 variables within each study.

- `y` point esimates of risk differences.
- `s2` sample variances of risk differences.
- `n1` sample sizes in treatment group 1 (brief alcohol interventions).
- `n2` sample sizes in treatment group 2 (control).
- `r1` event counts in treatment group 1.
- `r2` event counts in treatment group 2.
- `n` total sample sizes.

Source

Kaner EF, Dickinson HO, Beyer FR, Campbell F, Schlesinger C, Heather N, Saunders JB, Burnand B, Pienaar ED (2007). "Effectiveness of brief alcohol interventions in primary care populations." *Cochrane Database of Systematic Reviews*, **2**, Art. No.: CD004148. <doi: 10.1002/14651858.CD004148.pub3>

dat.lcj *A Meta-Analysis on the Effect of Progressive Resistance Strength Training Exercise vs. Control*

Description

This meta-analysis serves as an example to illustrate function usage in the package **altmeta**.

Usage

```
data("dat.lcj")
```

Format

A data frame containing 33 studies with the observed effect sizes and their within-study variances.

y the observed effect size for each collected study in the meta-analysis.

s2 the within-study variance for each study.

Source

Liu CJ, Latham NK (2009). "Progressive resistance strength training for improving physical function in older adults." *Cochrane Database of Systematic Reviews*, **3**. Art. No.: CD002759. <doi: 10.1002/14651858.CD002759.pub2>

---

dat.paige — *A Meta-Analysis on the Effectiveness of Spinal Manipulative Therapies (Other Than Sham)*

Description

This dataset serves as an example of meta-analysis of standardized mean differences.

Usage

```
data("dat.paige")
```

Format

A data frame with 6 studies with the following 5 variables within each study.

y point esimates of standardized mean differences.

s2 sample variances of standardized mean differences.

n1 sample sizes in treatment group 1 (spinal manipulation).

n2 sample sizes in treatment group 2 (comparator).

n total sample sizes.

Source

Paige NM, Miake-Lye IM, Booth MS, Beroes JM, Mardian AS, Dougherty P, Branson R, Tang B, Morton SC, Shekelle PG (2017). "Association of spinal manipulative therapy with clinical benefit and harm for acute low back pain: systematic review and meta-analysis." *JAMA*, **317**(14), 1451–1460. <doi: 10.1001/jama.2017.3086>

`dat.plourde` *A Meta-Analysis for Comparing the Fluoroscopy Time in Percutaneous Coronary Intervention Between Radial and Femoral Accesses*

Description

This dataset serves as an example of meta-analysis of mean differences.

Usage

```
data("dat.plourde")
```

Format

A data frame with 19 studies with the following 5 variables within each study.

- `y` point esimates of mean differences.
- `s2` sample variances of mean differences.
- `n1` sample sizes in treatment group 1 (radial).
- `n2` sample sizes in treatment group 2 (femoral).
- `n` total sample sizes.

Source

Plourde G, Pancholy SB, Nolan J, Jolly S, Rao SV, Amhed I, Bangalore S, Patel T, Dahm JB, Bertrand OF (2015). "Radiation exposure in relation to the arterial access site used for diagnostic coronary angiography and percutaneous coronary intervention: a systematic review and meta-analysis." *Lancet*, **386**(10009), 2192–2203. <doi: 10.1016/S01406736(15)003050>

`dat.poole` *A Meta-Analysis for Evaluating the Effect of Mucolytic on Bronchitis/Chronic Obstructive Pulmonary Disease*

Description

This meta-analysis serves as an example of meta-analysis with binary outcomes.

Usage

```
data("dat.poole")
```

Format

A data frame containing 24 studies with the following 4 variables within each study.

`n00` counts of non-events in treatment group 0 (placebo).

`n01` counts of events in treatment group 0 (placebo).

`n10` counts of non-events in treatment group 1 (mucolytic).

`n11` counts of events in treatment group 1 (mucolytic).

Source

Poole P, Chong J, Cates CJ (2015). "Mucolytic agents versus placebo for chronic bronchitis or chronic obstructive pulmonary disease." *Cochrane Database of Systematic Reviews*, **7**, Art. No.: CD001287. <doi: 10.1002/14651858.CD001287.pub5>

`dat.pte` *Meta-Analysis of Multiple Risk Factors for Pterygium*

Description

This dataset serves as an example to illustrate network meta-analysis of multiple factors. It consists of 29 studies on a total of 8 risk factors: area of residence (rural vs. urban); education attainment (low vs. high); latitude of residence (low vs. high); occupation type (outdoor vs. indoor); smoking status (yes vs. no); use of hat (yes vs. no); use of spectacles (yes vs. no); and use of sunglasses (yes vs. no). Each study only investigates a subset of the 8 risk factors, so the dataset contains many missing values.

Usage

```
data("dat.pte")
```

Format

A list containing two elements, `y` and `se`.

`y` a 29 x 8 numeric matrix containing the observed effect sizes; the rows represent studies and the columns represent outcomes.

`se` a 29 x 8 numeric matrix containing the within-study standard errors; the rows represent studies and the columns represent outcomes.

Source

Serghiou S, Patel CJ, Tan YY, Koay P, Ioannidis JPA (2016). "Field-wide meta-analyses of observational associations can map selective availability of risk factors and the impact of model specifications." *Journal of Clinical Epidemiology*, **71**, 58–67. <doi: 10.1016/j.jclinepi.2015.09.004>

`dat.slf` *A Meta-Analysis on the Effect of Nicotine Gum for Smoking Cessation*

Description

This meta-analysis serves as an example to illustrate function usage in the package **altmeta**.

Usage

```
data("dat.slf")
```

Format

A data frame containing 56 studies with the observed effect sizes and their within-study variances.

`y` the observed effect size for each collected study in the meta-analysis.

`s2` the within-study variance for each study.

Source

Stead LF, Perera R, Bullen C, Mant D, Hartmann-Boyce J, Cahill K, Lancaster T (2012). "Nicotine replacement therapy for smoking cessation." *Cochrane Database of Systematic Reviews*, **11**. Art. No.: CD000146. <doi: 10.1002/14651858.CD000146.pub4>

`dat.whiting` *A Meta-Analysis on Adverse Events for the Comparison Cannabinoid vs. Placebo*

Description

This dataset serves as an example of meta-analysis of (log) odds ratios.

Usage

```
data("dat.whiting")
```

Format

A data frame with 29 studies with the following 9 variables within each study.

`y` point esimates of log odds ratios.

`s2` sample variances of log odds ratios.

`n00` counts of non-events in treatment group 0 (placebo).

`n01` counts of events in treatment group 0.

`n10` counts of non-events in treatment group 1 (cannabinoid).

`n11` counts of events in treatment group 1 (cannabinoid).

`n0` sample sizes in treatment group 0.

`n1` sample sizes in treatment group 1.

`n` total sample sizes.

Source

Whiting PF, Wolff RF, Deshpande S, Di Nisio M, Duffy S, Hernandez AV, Keurentjes JC, Lang S, Misso K, Ryder S, Schmidlkofer S, Westwood M, Kleijnen J (2015). "Cannabinoids for medical use: a systematic review and meta-analysis." *JAMA*, **313**(24), 2456–2473. <doi: 10.1001/jama.2015.6358>

`dat.williams` *A Meta-Analysis on the Effect of Pharmacotherapy for Social Anxiety Disorder*

Description

This dataset serves as an example of meta-analysis of (log) relative risks.

Usage

```
data("dat.williams")
```

Format

A data frame with 20 studies with the following 7 variables within each study.

`y` point esimates of log relative risks.

`s2` sample variances of log relative risks.

`n1` sample sizes in treatment group 1 (medication).

`n2` sample sizes in treatment group 2 (placebo).

`r1` event counts in treatment group 1.

`r2` event counts in treatment group 2.

`n` total sample sizes.

Source

Williams T, Hattingh CJ, Kariuki CM, Tromp SA, van Balkom AJ, Ipser JC, Stein DJ (2017). "Pharmacotherapy for social anxiety disorder (SAnD)." *Cochrane Database of Systematic Reviews*, **10**, Art. No.: CD001206. <doi: 10.1002/14651858.CD001206.pub3>

`maprop.glmm` *Meta-Analysis of Proportions Using Generalized Linear Mixed Models*

Description

Performs a meta-analysis of proportions using generalized linear mixed models (GLMMs) with various link functions.

Usage

```
maprop.glmm(e, n, data, link = "logit", alpha = 0.05,
            pop.avg = TRUE, int.approx = 10000, b.iter = 1000,
            seed = 1234)
```

Arguments

`e`	a numeric vector indicating the event counts in the collected studies.
`n`	a numeric vector indicating the sample sizes in the collected studies.
`data`	an optional data frame containing the meta-analysis dataset. If `data` is specified, the previous arguments, `e` and `n`, should be specified as their corresponding column names in `data`.
`link`	a character string specifying the link function used in the GLMM, which can be one of `"log"` (log link), `"logit"` (logit link, the default), `"probit"` (probit link), `"cauchit"` (cauchit link), and `"cloglog"` (complementary log-log link).
`alpha`	a numeric value indicating the statistical significance level.
`pop.avg`	a logical value indicating whether the population-averaged proportion and its confidence interval are to be produced. This quantity is the marginal mean of study-specific proportions, while the commonly-reported overall proportion usually represents the median (or interpreted as a conditional measure); see more details about this quantity in Section 13.2.3 in Agresti (2013), Chu et al. (2012), Lin and Chu (2020), and Zeger et al. (1988). If `pop.avg = TRUE` (the default), the bootstrap resampling is used to produce the confidence interval of the population-averaged proportion; the confidence interval of the commonly-reported median proportion will be also produced, in addition to its conventional confidence interval (by back-transforming the Wald-type confidence interval derived on the scale specified by `link`).
`int.approx`	an integer indicating the number of independent standard normal samples for numerically approximating the integration involved in the calculation of the population-averaged proportion; see details in Lin and Chu (2020). It is only used when `pop.avg = TRUE` and `link` is not `"probit"`. The probit link leads to a closed form of the population-averaged proportion, so it does not need the numerical approximation; for other links, the population-averaged proportion does not have a closed form.

b.iter	an integer indicating the number of bootstrap iterations; it is only used when pop.avg = TRUE.
seed	an integer for specifying the seed of the random number generation for reproducibility during the bootstrap resampling (and numerical approximation for the population-averaged proportion); it is only used when pop.avg = TRUE.

Value

This function returns a list containing the point and interval estimates of the overall proportion. Specifically, prop.c.est is the commonly-reported median (or conditional) proportion, and prop.c.ci is its confidence interval. It also returns information about AIC, BIC, log likelihood, deviance, and residual degrees-of-freedom. If pop.avg = TRUE, the following additional elements will be also in the produced list: prop.c.ci.b is the bootstrap confidence interval of the commonly-reported median (conditional) proportion, prop.m.est is the point estimate of the population-averaged (marginal) proportion, prop.m.ci.b is the bootstrap confidence interval of the population-averaged (marginal) proportion, and b.w.e is a vector of two numeric values, indicating the counts of warnings and errors occurred during the bootstrap iterations.

Note

This function implements the GLMM for the meta-analysis of proportions via the glmer function in the package **lme4**. It is possible that the algorithm of the GLMM estimation may not converge for some bootstrapped meta-analyses when pop.avg = TRUE, and the glmer function may report warnings or errors about the convergence issue. The bootstrap iterations are continued until b.iter replicates without any warnings or errors are obtained; those replicates with any warnings or errors are discarded.

References

Agresti A (2013). *Categorical Data Analysis*. Third edition. John Wiley & Sons, Hoboken, NJ.

Bakbergenuly I, Kulinskaya E (2018). "Meta-analysis of binary outcomes via generalized linear mixed models: a simulation study." *BMC Medical Research Methodology*, **18**, 70. <doi: 10.1186/s1287401805319>

Chu H, Nie L, Chen Y, Huang Y, Sun W (2012). "Bivariate random effects models for meta-analysis of comparative studies with binary outcomes: methods for the absolute risk difference and relative risk." *Statistical Methods in Medical Research*, **21**(6), 621–633. <doi: 10.1177/0962280210393712>

Hamza TH, van Houwelingen HC, Stijnen T (2008). "The binomial distribution of meta-analysis was preferred to model within-study variability." *Journal of Clinical Epidemiology*, **61**(1), 41–51. <doi: 10.1016/j.jclinepi.2007.03.016>

Lin L, Chu H (2020). "Meta-analysis of proportions using generalized linear mixed models." *Epidemiology*, in press. <doi: 10.1097/ede.0000000000001232>

Stijnen T, Hamza TH, Ozdemir P (2010). "Random effects meta-analysis of event outcome in the framework of the generalized linear mixed model with applications in sparse data." *Statistics in Medicine*, **29**(29), 3046–3067. <doi: 10.1002/sim.4040>

Zeger SL, Liang K-Y, Albert PS (1988). "Models for longitudinal data: a generalized estimating equation approach." *Biometrics*, **44**(4), 1049–1060. <doi: 10.2307/2531734>

See Also

maprop.twostep

Examples

```
# chorioamnionitis data
data("dat.chor")
# GLMM with the logit link with only 10 bootstrap iterations
out.chor.glmm.logit <- maprop.glmm(e, n, data = dat.chor,
  link = "logit", b.iter = 10, seed = 1234)
out.chor.glmm.logit
# not calculating the population-averaged (marginal) proportion,
#  without bootstrap resampling
out.chor.glmm.logit <- maprop.glmm(e, n, data = dat.chor,
  link = "logit", pop.avg = FALSE)
out.chor.glmm.logit

# increases the number of bootstrap iterations to 1000,
#  taking longer time
out.chor.glmm.logit <- maprop.glmm(e, n, data = dat.chor,
  link = "logit", b.iter = 1000, seed = 1234)
out.chor.glmm.logit

# GLMM with the log link
out.chor.glmm.log <- maprop.glmm(e, n, data = dat.chor,
  link = "log", b.iter = 10, seed = 1234)
out.chor.glmm.log
# GLMM with the probit link
out.chor.glmm.probit <- maprop.glmm(e, n, data = dat.chor,
  link = "probit", b.iter = 10, seed = 1234)
out.chor.glmm.probit
# GLMM with the cauchit link
out.chor.glmm.cauchit <- maprop.glmm(e, n, data = dat.chor,
  link = "cauchit", b.iter = 10, seed = 1234)
out.chor.glmm.cauchit
# GLMM with the cloglog link
out.chor.glmm.cloglog <- maprop.glmm(e, n, data = dat.chor,
  link = "cloglog", b.iter = 10, seed = 1234)
out.chor.glmm.cloglog


# depression data
data("dat.beck17")
out.beck17.glmm.log <- maprop.glmm(e, n, data = dat.beck17,
  link = "log", b.iter = 10, seed = 1234)
out.beck17.glmm.log
out.beck17.glmm.logit <- maprop.glmm(e, n, data = dat.beck17,
  link = "logit", b.iter = 10, seed = 1234)
out.beck17.glmm.logit
out.beck17.glmm.probit <- maprop.glmm(e, n, data = dat.beck17,
  link = "probit", b.iter = 10, seed = 1234)
out.beck17.glmm.probit
```

```
out.beck17.glmm.cauchit <- maprop.glmm(e, n, data = dat.beck17,
  link = "cauchit", b.iter = 10, seed = 1234)
out.beck17.glmm.cauchit
out.beck17.glmm.cloglog<- maprop.glmm(e, n, data = dat.beck17,
  link = "cloglog", b.iter = 10, seed = 1234)
out.beck17.glmm.cloglog
```

---

maprop.twostep *Meta-Analysis of Proportions Using Two-Step Methods*

Description

Performs a meta-analysis of proportions using conventional two-step methods with various data transformations.

Usage

```
maprop.twostep(e, n, data, link = "logit", method = "ML", alpha = 0.05,
               pop.avg = TRUE, int.approx = 10000, b.iter = 1000,
               seed = 1234)
```

Arguments

`e`	a numeric vector indicating the event counts in the collected studies.
`n`	a numeric vector indicating the sample sizes in the collected studies.
`data`	an optional data frame containing the meta-analysis dataset. If `data` is specified, the previous arguments, `e` and `n`, should be specified as their corresponding column names in `data`.
`link`	a character string specifying the data transformation for each study's proportion used in the two-step method, which can be one of `"log"` (log transformation), `"logit"` (logit transformation, the default), `"arcsine"` (arcsine transformation), and `"double.arcsine"` (Freeman–Tukey double-arcsine transformation).
`method`	a character string specifying the method to perform the meta-analysis, which is passed to the argument `method` in the function `rma.uni` in the package **metafor**. It can be one of `"ML"` (maximum likelihood, the default), `"REML"` (restricted maximum likelihood), and many other options; see more details in the manual of **metafor**. The default is set to `"ML"` for consistency with the function `maprop.glmm`, where generalized linear mixed models are often estimated via the maximum likelihood approach. For the two-step method, users might also use `"REML"` because the restricted maximum likelihood estimation may have superior performance in many cases.
`alpha`	a numeric value indicating the statistical significance level.

`pop.avg` a logical value indicating whether the population-averaged proportion and its confidence interval are to be produced. This quantity is the marginal mean of study-specific proportions, while the commonly-reported overall proportion usually represents the median (or interpreted as a conditional measure); see more details about this quantity in Section 13.2.3 in Agresti (2013), Chu et al. (2012), Lin and Chu (2020), and Zeger et al. (1988). If `pop.avg = TRUE` (the default), the bootstrap resampling is used to produce the confidence interval of the population-averaged proportion; the confidence interval of the commonly-reported median proportion will be also produced, in addition to its conventional confidence interval (by back-transforming the Wald-type confidence interval derived on the scale specified by `link`).

`int.approx` an integer indicating the number of independent standard normal samples for numerically approximating the integration involved in the calculation of the population-averaged proportion; see details in Lin and Chu (2020). It is only used when `pop.avg = TRUE`. For the commonly-used data transformations available for `link`, the population-averaged proportion does not have a closed form.

`b.iter` an integer indicating the number of bootstrap iterations; it is only used when `pop.avg = TRUE`.

`seed` an integer for specifying the seed of the random number generation for reproducibility during the bootstrap resampling (and numerical approximation for the population-averaged proportion); it is only used when `pop.avg = TRUE`.

Value

This function returns a list containing the point and interval estimates of the overall proportion. Specifically, `prop.c.est` is the commonly-reported median (or conditional) proportion, and `prop.c.ci` is its confidence interval. If `pop.avg = TRUE`, the following additional elements will be also in the produced list: `prop.c.ci.b` is the bootstrap confidence interval of the commonly-reported median (conditional) proportion, `prop.m.est` is the point estimate of the population-averaged (marginal) proportion, `prop.m.ci.b` is the bootstrap confidence interval of the population-averaged (marginal) proportion, and `b.w.e` is a vector of two numeric values, indicating the counts of warnings and errors occurred during the bootstrap iterations. Moreover, if the Freeman–Tukey double-arcsine transformation (`link = "double.arcsine"`) is used, the back-transformation will be implemented at four values as the overall sample size: the harmonic, geometric, and arithmetic means of the study-specific sample sizes, and the inverse of the synthesized result's variance. See details in Barendregt et al. (2013) and Schwarzer et al. (2019).

Note

This function implements the two-step method for the meta-analysis of proportions via the `rma.uni` function in the package **metafor**. It is possible that the algorithm of the maximum likelihood or restricted maximum likelihood estimation may not converge for some bootstrapped meta-analyses when `pop.avg = TRUE`, and the `rma.uni` function may report warnings or errors about the convergence issue. The bootstrap iterations are continued until `b.iter` replicates without any warnings or errors are obtained; those replicates with any warnings or errors are discarded.

References

Agresti A (2013). *Categorical Data Analysis*. Third edition. John Wiley & Sons, Hoboken, NJ.

Barendregt JJ, Doi SA, Lee YY, Norman RE, Vos T (2013). "Meta-analysis of prevalence." *Journal of Epidemiology and Community Health*, **67**(11), 974–978. <doi: 10.1136/jech2013203104>

Freeman MF, Tukey JW (1950). "Transformations related to the angular and the square root." *The Annals of Mathematical Statistics*, **21**(4), 607–611. <doi: 10.1214/aoms/1177729756>

Lin L, Chu H (2020). "Meta-analysis of proportions using generalized linear mixed models." *Epidemiology*, in press. <doi: 10.1097/ede.0000000000001232>

Miller JJ (1978). "The inverse of the Freeman–Tukey double arcsine transformation." *The American Statistician*, **32**(4), 138. <doi: 10.1080/00031305.1978.10479283>

Schwarzer G, Chemaitelly H, Abu-Raddad LJ, Rucker G (2019). "Seriously misleading results using inverse of Freeman-Tukey double arcsine transformation in meta-analysis of single proportions." *Research Synthesis Methods*, **10**(3), 476–483. <doi: 10.1002/jrsm.1348>

Viechtbauer W (2010). "Conducting meta-analyses in R with the metafor package." *Journal of Statistical Software*, **36**, 3. <doi: 10.18637/jss.v036.i03>

Zeger SL, Liang K-Y, Albert PS (1988). "Models for longitudinal data: a generalized estimating equation approach." *Biometrics*, **44**(4), 1049–1060. <doi: 10.2307/2531734>

See Also

maprop.glmm

Examples

```
# chorioamnionitis data
data("dat.chor")
# two-step method with the logit transformation
out.chor.twostep.logit <- maprop.twostep(e, n, data = dat.chor,
  link = "logit", b.iter = 10, seed = 1234)
out.chor.twostep.logit
# not calculating the population-averaged (marginal) proportion,
#  without bootstrap resampling
out.chor.twostep.logit <- maprop.twostep(e, n, data = dat.chor,
  link = "logit", pop.avg = FALSE)
out.chor.twostep.logit

# increases the number of bootstrap iterations to 1000,
#  taking longer time
out.chor.twostep.logit <- maprop.twostep(e, n, data = dat.chor,
  link = "logit", b.iter = 1000, seed = 1234)
out.chor.twostep.logit

# two-step method with the log transformation
out.chor.twostep.log <- maprop.twostep(e, n, data = dat.chor,
  link = "log", b.iter = 10, seed = 1234)
out.chor.twostep.log
# two-step method with the arcsine transformation
out.chor.twostep.arcsine <- maprop.twostep(e, n, data = dat.chor,
  link = "arcsine", b.iter = 10, seed = 1234)
out.chor.twostep.arcsine
# two-step method with the Freeman--Tukey double-arcsine transformation
```

```
out.chor.twostep.double.arcsine <- maprop.twostep(e, n, data = dat.chor,
  link = "double.arcsine", b.iter = 10, seed = 1234)
out.chor.twostep.double.arcsine


# depression data
data("dat.beck17")
out.beck17.twostep.log <- maprop.twostep(e, n, data = dat.beck17,
  link = "log", b.iter = 10, seed = 1234)
out.beck17.twostep.log
out.beck17.twostep.logit <- maprop.twostep(e, n, data = dat.beck17,
  link = "logit", b.iter = 10, seed = 1234)
out.beck17.twostep.logit
out.beck17.twostep.arcsine <- maprop.twostep(e, n, data = dat.beck17,
  link = "arcsine", b.iter = 10, seed = 1234)
out.beck17.twostep.arcsine
out.beck17.twostep.double.arcsine <- maprop.twostep(e, n, data = dat.beck17,
  link = "double.arcsine", b.iter = 10, seed = 1234)
out.beck17.twostep.double.arcsine
```

`metahet` *Meta-Analysis Heterogeneity Measures*

Description

Calculates various between-study heterogeneity measures in meta-analysis, including the conventional measures (e.g., I^2) and the alternative measures (e.g., I_r^2) which are robust to outlying studies; p-values of various tests are also calculated.

Usage

```
metahet(y, s2, n.resam = 1000)
```

Arguments

`y`	a numeric vector indicating the observed effect sizes in the collected studies; they are assumed to be normally distributed.
`s2`	a numeric vector indicating the within-study variances.
`n.resam`	a positive integer indicating the number of resampling iterations for calculating p-values of test statistics and 95% confidence interval of heterogeneity measures.

Details

Suppose that a meta-analysis collects n studies. The observed effect size in study i is y_i and its within-study variance is s_i^2. Also, the inverse-variance weight is $w_i = 1/s_i^2$. The fixed-effect estimate of overall effect size is $\bar{\mu} = \sum_{i=1}^{n} w_i y_i / \sum_{i=1}^{n} w_i$. The conventional test statistic for heterogeneity is

$$Q = \sum_{i=1}^{n} w_i (y_i - \bar{\mu})^2.$$

Based on the Q statistic, the method-of-moments estimate of the between-study variance $\hat{\tau}_{DL}^2$ is (DerSimonian and Laird, 1986)

$$\hat{\tau}_{DL}^2 = \max \left\{ 0, \frac{Q - (n-1)}{\sum_{i=1}^{n} w_i - \sum_{i=1}^{n} w_i^2 / \sum_{i=1}^{n} w_i} \right\}.$$

Also, the H and I^2 statistics (Higgins and Thompson, 2002; Higgins et al., 2003) are widely used in practice because they do not depend on the number of collected studies n and the effect size scale; these two statistics are defined as

$$H = \sqrt{Q/(n-1)};$$
$$I^2 = \frac{Q - (n-1)}{Q}.$$

Specifically, the H statistic reflects the ratio of the standard deviation of the underlying mean from a random-effects meta-analysis compared to the standard deviation from a fixed-effect meta-analysis; the I^2 statistic describes the proportion of total variance across studies that is due to heterogeneity rather than sampling error.

Outliers are frequently present in meta-analyses, and they may have great impact on the above heterogeneity measures. Alternatively, to be more robust to outliers, the test statistic may be modified as (Lin et al., 2017):

$$Q_r = \sum_{i=1}^{n} \sqrt{w_i} |y_i - \bar{\mu}|.$$

Based on the Q_r statistic, the method-of-moments estimate of the between-study variance $\hat{\tau}_r^2$ is defined as the solution to

$$Q_r \sqrt{\frac{\pi}{2}} = \sum_{i=1}^{n} \left\{ 1 - \frac{w_i}{\sum_{j=1}^{n} w_j} + \tau^2 \left[w_i - \frac{2 w_i^2}{\sum_{j=1}^{n} w_j} + \frac{w_i \sum_{j=1}^{n} w_j^2}{(\sum_{j=1}^{n} w_j)^2} \right] \right\}.$$

If no positive solution exists to the equation above, set $\hat{\tau}_r^2 = 0$. The counterparts of the H and I^2 statistics are defined as

$$H_r = Q_r \sqrt{\pi / [2n(n-1)]};$$
$$I_r^2 = \frac{Q_r^2 - 2n(n-1)/\pi}{Q_r^2}.$$

To further improve the robustness of heterogeneity assessment, the weighted *mean* in the Q_r statistic may be replaced by the weighted *median* $\hat{\mu}_m$, which is the solution to $\sum_{i=1}^{n} w_i [I(\theta \geq y_i) - 0.5] = 0$ with respect to θ. The new test statistic is

$$Q_m = \sum_{i=1}^{n} \sqrt{w_i} |y_i - \hat{\mu}_m|.$$

Based on Q_m, the new estimator of the between-study variance $\hat{\tau}_m^2$ is the solution to

$$Q_m \sqrt{\pi/2} = \sum_{i=1}^{n} \sqrt{(s_i^2 + \tau^2)/s_i^2}.$$

The counterparts of the H and I^2 statistics are

$$H_m = \frac{Q_m}{n}\sqrt{\pi/2};$$

$$I_m^2 = \frac{Q_m^2 - 2n^2/\pi}{Q_m^2}.$$

Value

This function returns a list containing p-values of various heterogeneity tests and various heterogeneity measures with 95% confidence intervals. Specifically, the components include:

`p.Q`	p-value of the Q statistic (using the resampling method).
`p.Q.theo`	p-value of the Q statistic using the Q's theoretical chi-squared distribution.
`p.Qr`	p-value of the Q_r statistic (using the resampling method).
`p.Qm`	p-value of the Q_m statistic (using the resampling method).
`Q`	the Q statistic.
`ci.Q`	95% CI of the Q statistic.
`tau2.DL`	DerSimonian–Laird estimate of the between-study variance.
`ci.tau2.DL`	95% CI of the between-study variance based on the DerSimonian–Laird method.
`H`	the H statistic.
`ci.H`	95% CI of the H statistic.
`I2`	the I^2 statistic.
`ci.I2`	95% CI of the I^2 statistic.
`Qr`	the Q_r statistic.
`ci.Qr`	95% CI of the Q_r statistic.
`tau2.r`	the between-study variance estimate based on the Q_r statistic.
`ci.tau2.r`	95% CI of the between-study variance based on the Q_r statistic.
`Hr`	the H_r statistic.
`ci.Hr`	95% CI of the H_r statistic.
`Ir2`	the I_r^2 statistic.
`ci.Ir2`	95% CI of the I_r^2 statistic.
`Qm`	the Q_m statistic.
`ci.Qm`	95% CI of the Q_m statistic.
`tau2.m`	the between-study variance estimate based on the Q_m statistic.
`ci.tau2.m`	95% CI of the between-study variance based on the Q_m statistic
`Hm`	the H_m statistic.
`ci.Hm`	95% CI of the H_m statistic.
`Im2`	the I_m^2 statistic.
`ci.Im2`	95% CI of the I_m^2 statistic.

References

DerSimonian R, Laird N (1986). "Meta-analysis in clinical trials." *Controlled Clinical Trials*, **7**(3), 177–188. <doi: 10.1016/01972456(86)900462>

Higgins JPT, Thompson SG (2002). "Quantifying heterogeneity in a meta-analysis." *Statistics in Medicine*, **21**(11), 1539–1558. <doi: 10.1002/sim.1186>

Higgins JPT, Thompson SG, Deeks JJ, Altman DG (2003). "Measuring inconsistency in meta-analyses." *BMJ*, **327**(7414), 557–560. <doi: 10.1136/bmj.327.7414.557>

Lin L, Chu H, Hodges JS (2017). "Alternative measures of between-study heterogeneity in meta-analysis: reducing the impact of outlying studies." *Biometrics*, **73**(1), 156–166. <doi: 10.1111/biom.12543>

Examples

```
data("dat.aex")
set.seed(1234)
attach(dat.aex)
metahet(y, s2, 100)
metahet(y, s2, 1000)
detach(dat.aex)

data("dat.hipfrac")
set.seed(1234)
attach(dat.hipfrac)
metahet(y, s2, 100)
metahet(y, s2, 1000)
detach(dat.hipfrac)
```

`metaoutliers` *Outlier Detection in Meta-Analysis*

Description

Calculates the standardized residual for each study in meta-analysis using the methods desribed in Chapter 12 in Hedges and Olkin (1985) and Viechtbauer and Cheung (2010). A study is considered as an outlier if its standardized residual is greater than 3 in absolute magnitude.

Usage

```
metaoutliers(y, s2, model)
```

Arguments

`y`	a numeric vector indicating the observed effect sizes in the collected studies; they are assumed to be normally distributed.
`s2`	a numeric vector indicating the within-study variances.

`model` a character string specified as either "FE" or "RE". If `model = "FE"`, this function uses the outlier detection procedure for the fixed-effect meta-analysis desribed in Chapter 12 in Hedges and Olkin (1985); If `model = "RE"`, the procedure for the random-effects meta-analysis desribed in Viechtbauer and Cheung (2010) is used. See Details for the two approaches. If the argument `model` is not specified, this function sets `model = "FE"` if $I_r^2 < 30\%$ and sets `model = "RE"` if $I_r^2 \geq 30\%$.

Details

Suppose that a meta-analysis collects n studies. The observed effect size in study i is y_i and its within-study variance is s_i^2. Also, the inverse-variance weight is $w_i = 1/s_i^2$.

Chapter 12 in Hedges and Olkin (1985) describes the outlier detection procedure for the fixed-effect meta-analysis (`model = "FE"`). Using the studies except study i, the pooled estimate of the overall effect size is $\bar{\mu}_{(-i)} = \sum_{j \neq i} w_j y_j / \sum_{j \neq i} w_j$. The residual of study i is $e_i = y_i - \bar{\mu}_{(-i)}$. The variance of e_i is $v_i = s_i^2 + (\sum_{j \neq i} w_j)^{-1}$, so the standardized residual of study i is $\epsilon_i = e_i/\sqrt{v_i}$.

Viechtbauer and Cheung (2010) describes the outlier detection procedure for the random-effects meta-analysis (`model = "RE"`). Using the studies except study i, let the method-of-moments estimate of the between-study variance be $\hat{\tau}^2_{(-i)}$. The pooled estimate of the overall effect size is $\bar{\mu}_{(-i)} = \sum_{j \neq i} \tilde{w}_{(-i)j} y_j / \sum_{j \neq i} \tilde{w}_{(-i)j}$, where $\tilde{w}_{(-i)j} = 1/(s_j^2 + \hat{\tau}^2_{(-i)})$. The residual of study i is $e_i = y_i - \bar{\mu}_{(-i)}$, and its variance is $v_i = s_i^2 + \hat{\tau}^2_{(-i)} + (\sum_{j \neq i} \tilde{w}_{(-i)j})^{-1}$. Then, the standardized residual of study i is $\epsilon_i = e_i/\sqrt{v_i}$.

Value

This functions returns a list which contains standardized residuals and identified outliers. A study is considered as an outlier if its standardized residual is greater than 3 in absolute magnitude.

References

Hedges LV, Olkin I (1985). *Statistical Method for Meta-Analysis*. Academic Press, Orlando, FL.

Viechtbauer W, Cheung MWL (2010). "Outlier and influence diagnostics for meta-analysis." *Research Synthesis Methods*, **1**(2), 112–125. <doi: 10.1002/jrsm.11>

Examples

```
data("dat.aex")
attach(dat.aex)
metaoutliers(y, s2, model = "FE")
metaoutliers(y, s2, model = "RE")
detach(dat.aex)

data("dat.hipfrac")
attach(dat.hipfrac)
metaoutliers(y, s2)
detach(dat.hipfrac)
```

`metapb` *Detecting and Quantifying Publication Bias*

Description

Performs the regression test and calculates skewness for detecting and quantifying publication bias.

Usage

```
metapb(y, s2, model)
```

Arguments

`y`	a numeric vector indicating the observed effect sizes in the collected studies; they are assumed to be normally distributed.
`s2`	a numeric vector indicating the within-study variances.
`model`	a characher string specifying the fixed-effect (`"FE"`) or random-effects (`"RE"`) model. If not specified, this function uses the Q statistic to test for heterogeneity: if the p-value is smaller than 0.05, `model` is set to `"RE"`; otherwise, `model = "FE"`.

Details

This function derives the measures of publication bias introduced in Lin and Chu (2018).

Value

This function returns a list containing measures of publication bias, their 95% confidence intervals, and p-values. Specifically, the components include:

`n`	the number of studies in the meta-analysis.
`p.Q`	the p-value of the Q-test for heterogeneity.
`I2`	the I^2 statistic for quantifying heterogeneity.
`tau2`	the DerSimonian–Laird estimate of the between-study variance.
`model`	the model setting (`"FE"` or `"RE"`).
`std.dev`	the standardized deviates of the studies.
`reg.int`	the estimate of the regression intercept for quantifying publication bias.
`reg.int.ci`	the 95% CI of the regression intercept.
`reg.pval`	the p-value of the regression intercept.
`skewness`	the estimate of the skewness for quantifying publication bias.
`skewness.ci`	the 95% CI of the skewness.
`skewness.pval`	the p-value of the skewness.
`combined.pval`	the p-value of the combined test that incorporates the regression intercept and the skewness.

References

Egger M, Davey Smith G, Schneider M, Minder C (1997). "Bias in meta-analysis detected by a simple, graphical test." *BMJ*, **315**(7109), 629–634. <doi: 10.1136/bmj.315.7109.629>

Lin L, Chu H (2018). "Quantifying publication bias in meta-analysis." *Biometrics*, **74**(3), 785–794. <doi: 10.1111/biom.12817>

Examples

```
data("dat.slf")
attach(dat.slf)
metapb(y, s2)
detach(dat.slf)

data("dat.ha")
attach(dat.ha)
metapb(y, s2)
detach(dat.ha)

data("dat.lcj")
attach(dat.lcj)
metapb(y, s2)
detach(dat.lcj)
```

---

mvma *Multivariate Meta-Analysis*

Description

Performs a multivariate meta-analysis when the within-study correlations are known.

Usage

```
mvma(ys, covs, method = "reml", tol = 1e-10)
```

Arguments

`ys`	an n x p numeric matrix containing the observed effect sizes. The n rows represent studies, and the p columns represent the multivariate endpoints. `NA` is allowed for missing endpoints.
`covs`	a numeric list with length n. Each element is the p x p within-study covariance matrix. `NA` is allowed for missing endpoints in the covariance matrix.
`method`	a character string indicating the method for estimating the overall effect sizes. It should be `"fe"` (fixed-effects model), `"ml"` (random-effects model using the maximum likelihood method), or `"reml"` (random-effects model using the restricted maximum likelihood method, the default).
`tol`	a small number indicating the convergence tolerance for the estimates by maximizing (restricted) likelihood. The default is `1e-10`.

Details

Suppose n studies are collected in a multivariate meta-analysis on a total of p endpoints. Denote the p-dimensional vector of effect sizes as $\boldsymbol{y}_i$, and the within-study covariance matrix $\mathbf{S}_i$ is assumed to be known. Then, the random-effects model is as follows:

$$\boldsymbol{y}_i \sim N(\boldsymbol{\mu}_i, \mathbf{S}_i);$$

$$\boldsymbol{\mu}_i \sim N(\boldsymbol{\mu}, \mathbf{T}).$$

Here, $\boldsymbol{\mu}_i$ represents the true underlying effect sizes in study i, $\boldsymbol{\mu}$ represents the overall effect sizes across studies, and $\mathbf{T}$ is the between-study covariance matrix due to heterogeneity. By setting $\mathbf{T} = \mathbf{0}$, this model becomes the fixed-effects model.

Value

This function returns a list containing the following elements:

`mu.est`	The estimated overall effect sizes of the p endpoints.
`Tau.est`	The estimated between-study covariance matrix.
`mu.cov`	The covariance matrix of the estimated overall effect sizes.
`method`	The method used to produce the estimates.

References

Jackson D, Riley R, White IR (2011). "Multivariate meta-analysis: potential and promise." *Statistics in Medicine*, **30**(20), 2481–2498. <doi: 10.1002/sim.4172>

See Also

`mvma.bayesian`, `mvma.hybrid`, `mvma.hybrid.bayesian`

Examples

```
data("dat.fib")
mvma(ys = dat.fib$y, covs = dat.fib$S, method = "fe")
mvma(ys = dat.fib$y, covs = dat.fib$S, method = "reml")
```

`mvma.bayesian` *Bayesian Random-Effects Multivariate Meta-Analysis*

Description

Performs a Bayesian random-effects model for multivariate meta-analysis when the within-study correlations are known.

Usage

```
mvma.bayesian(ys, covs, n.adapt = 1000, n.chains = 3,
              n.burnin = 10000, n.iter = 10000, n.thin = 1,
              data.name = NULL, traceplot = FALSE, coda = FALSE)
```

Arguments

`ys`	an n x p numeric matrix containing the observed effect sizes. The n rows represent studies, and the p columns represent the multivariate endpoints. `NA` is allowed for missing endpoints.
`covs`	a numeric list with length n. Each element is the p x p within-study covariance matrix. `NA` is allowed for missing endpoints in the covariance matrix.
`n.adapt`	the number of iterations for adaptation in the Markov chain Monte Carlo (MCMC) algorithm. The default is 1,000. This argument and the following `n.chains`, `n.burnin`, `n.iter`, and `n.thin` are passed to the functions in the package **rjags**.
`n.chains`	the number of MCMC chains. The default is 3.
`n.burnin`	the number of iterations for burn-in period. The default is 10,000.
`n.iter`	the total number of iterations in each MCMC chain after the burn-in period. The default is 10,000.
`n.thin`	a positive integer indicating thinning rate. The default is 1.
`data.name`	a character string indicating the data name. This is used in the names of the generated files that contain results. The default is `NULL`.
`traceplot`	a logical value indicating whether to save trace plots for the overall effect sizes and between-study standard deviations. The default is `FALSE`.
`coda`	a logical value indicating whether to output MCMC posterior samples. The default is `FALSE`.

Details

Suppose n studies are collected in a multivariate meta-analysis on a total of p endpoints. Denote the p-dimensional vector of effect sizes as $\boldsymbol{y}_i$, and the within-study covariance matrix $\mathbf{S}_i$ is assumed to be known. Then, the random-effects model is as follows:

$$\boldsymbol{y}_i \sim N(\boldsymbol{\mu}_i, \mathbf{S}_i);$$

$$\boldsymbol{\mu}_i \sim N(\boldsymbol{\mu}, \mathbf{T}).$$

Here, $\boldsymbol{\mu}_i$ represents the true underlying effect sizes in study i, $\boldsymbol{\mu}$ represents the overall effect sizes across studies, and $\mathbf{T}$ is the between-study covariance matrix due to heterogeneity.

The vague priors $N(0, 10^3)$ are specified for the fixed effects $\boldsymbol{\mu}$. Also, this function uses the separation strategy to specify vague priors for the variance and correlation components in $\mathbf{T}$ (Pinheiro and Bates, 1996); this technique is considered less sensitive to hyperparameters compared to specifying the inverse-Wishart prior (Lu and Ades, 2009; Wei and Higgins, 2013). Specifically, write the between-study covariance matrix as $\mathbf{T} = \mathbf{D}^{1/2}\mathbf{R}\mathbf{D}^{1/2}$, where the diagonal matrix $\mathbf{D} = diag(\mathbf{T}) = diag(\tau_1^2, \ldots, \tau_p^2)$ contains the between-study variances, and $\mathbf{R}$ is the correlation matrix. Uniform priors $U(0, 10)$ are specified for τ_j's ($j = 1, \ldots, p$). Further, the correlation matrix can be written as $\mathbf{R} = \mathbf{L}\mathbf{L}^T$, where $\mathbf{L} = (L_{ij})$ is a lower triangular matrix with nonnegative diagonal elements. Also, $L_{11} = 1$ and for $i = 2, \ldots, p$, $L_{ij} = \cos\theta_{i2}$ if $j = 1$; $L_{ij} = (\prod_{k=2}^{j} \sin\theta_{ik}) \cos\theta_{i,j+1}$ if $j = 2, \ldots, i-1$; and $L_{ij} = \prod_{k=2}^{i} \sin\theta_{ik}$ if $j = i$. Here, θ_{ij}'s are angle parameters for $2 \leq j \leq i \leq p$, and $\theta_{ij} \in (0, \pi)$. Uniform priors are specified for the angle parameters: $\theta_{ij} \sim U(0, \pi)$.

Value

This functions produces posterior estimates and Gelman and Rubin's potential scale reduction factor, and it generates several files that contain trace plots (if `traceplot = TRUE`) and MCMC posterior samples (if `coda = TRUE`) in users' working directory. In these results, `mu` represents the overall effect sizes, `tau` represents the between-study variances, `R` contains the elements of the correlation matrix, and `theta` represents the angle parameters (see "Details").

Note

This function only implements the MCMC algorithm for the random-effects multivariate model, but not the fixed-effects model. Generally, the fixed-effects model can be easily implemented using the function `mvma`. However, when using `mvma` to fit the random-effects model, a large number of parameters need to be estimated, and the algorithm for maximizing (restricted) likelihood may not converge well. The Bayesian method in this function provides an alternative.

If a warning "adaptation incomplete" appears, users may increase `n.adapt`.

References

Gelman A, Rubin DB (1992). "Inference from iterative simulation using multiple sequences." *Statistical Science*, **7**(4), 457–472. <doi: 10.1214/ss/1177011136>

Jackson D, Riley R, White IR (2011). "Multivariate meta-analysis: potential and promise." *Statistics in Medicine*, **30**(20), 2481–2498. <doi: 10.1002/sim.4172>

Lu G, Ades AE (2009). "Modeling between-trial variance structure in mixed treatment comparisons." *Biostatistics*, **10**(4), 792–805. <doi: 10.1093/biostatistics/kxp032>

Pinheiro JC, Bates DM (1996). "Unconstrained parametrizations for variance-covariance matrices." *Statistics and Computing*, **6**(3), 289–296. <doi: 10.1007/BF00140873>

Wei Y, Higgins JPT (2013). "Bayesian multivariate meta-analysis with multiple outcomes." *Statistics in Medicine*, **32**(17), 2911–2934. <doi: 10.1002/sim.5745>

See Also

`mvma`, `mvma.hybrid`, `mvma.hybrid.bayesian`

Examples

```
data("dat.fib")
set.seed(12345)
## increase n.burnin and n.iter for better convergence of MCMC
out <- mvma.bayesian(ys = dat.fib$y, covs = dat.fib$S,
  n.adapt = 1000, n.chains = 3, n.burnin = 100, n.iter = 100,
  n.thin = 1, data.name = "Fibrinogen")
out
```

`mvma.hybrid` *Hybrid Model for Random-Effects Multivariate Meta-Analysis*

Description

Performs a multivariate meta-analysis using the hybrid random-effects model when the within-study correlations are unknown.

Usage

```
mvma.hybrid(ys, vars, method = "reml", tol = 1e-10)
```

Arguments

`ys`	an n x p numeric matrix containing the observed effect sizes. The n rows represent studies, and the p columns represent the multivariate endpoints. `NA` is allowed for missing endpoints.
`vars`	an n x p numeric matrix containing the observed within-study variances. The n rows represent studies, and the p columns represent the multivariate endpoints. `NA` is allowed for missing endpoints.
`method`	a character string indicating the method for estimating the overall effect sizes. It should be `"ml"` (random-effects model using the maximum likelihood method) or `"reml"` (random-effects model using the restricted maximum likelihood method, the default).
`tol`	a small number indicating the convergence tolerance for the estimates by maximizing (restricted) likelihood. The default is `1e-10`.

Details

Suppose n studies are collected in a multivariate meta-analysis on a total of p endpoints. Denote the p-dimensional vector of effect sizes as $\boldsymbol{y}_i$, and their within-study variances form a diagonal matrix $\mathbf{D}_i$. However, the within-study correlations are unknown. Then, the random-effects hybrid model is as follows (Riley et al., 2008; Lin and Chu, 2018):

$$\boldsymbol{y}_i \sim N(\boldsymbol{\mu}, (\mathbf{D}_i + \mathbf{T})^{1/2}\mathbf{R}(\mathbf{D}_i + \mathbf{T})^{1/2}),$$

where $\boldsymbol{\mu}$ represents the overall effect sizes across studies, $\mathbf{T} = diag(\tau_1^2, \ldots, \tau_p^2)$ consists of the between-study variances, and $\mathbf{R}$ is the marginal correlation matrix. Although the within-study correlations are unknown, this model accounts for both within- and between-study correlations by using the marginal correlation matrix.

Value

This function returns a list containing the following elements:

`mu.est`	The estimated overall effect sizes of the p endpoints.
`tau2.est`	The estimated between-study variances of the p endpoints.
`mar.R`	The estimated marginal correlation matrix.
`mu.cov`	The covariance matrix of the estimated overall effect sizes.
`method`	The method used to produce the estimates.

Note

The algorithm for maximizing (restricted) likelihood may not converge when the dimension of endpoints is too high or the data are too sparse.

References

Lin L, Chu H (2018), "Bayesian multivariate meta-analysis of multiple factors." *Research Synthesis Methods*, **9**(2), 261–272. <doi: 10.1002/jrsm.1293>

Riley RD, Thompson JR, Abrams KR (2008), "An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown." *Biostatistics*, **9**(1), 172–186. <doi: 10.1093/biostatistics/kxm023>

See Also

`mvma`, `mvma.bayesian`, `mvma.hybrid.bayesian`

Examples

```
data("dat.fib")
y <- dat.fib$y
sd <- dat.fib$sd
mvma.hybrid(y = y, vars = sd^2)
```

---

mvma.hybrid.bayesian *Bayesian Hybrid Model for Random-Effects Multivariate Meta-Analysis*

Description

Performs a multivariate meta-analysis using the Bayesian hybrid random-effects model when the within-study correlations are unknown.

Usage

```
mvma.hybrid.bayesian(ys, vars, n.adapt = 1000, n.chains = 3,
                     n.burnin = 10000, n.iter = 10000, n.thin = 1,
                     data.name = NULL, traceplot = FALSE, coda = FALSE)
```

Arguments

`ys`	an n x p numeric matrix containing the observed effect sizes. The n rows represent studies, and the p columns represent the multivariate endpoints. `NA` is allowed for missing endpoints.
`vars`	an n x p numeric matrix containing the observed within-study variances. The n rows represent studies, and the p columns represent the multivariate endpoints. `NA` is allowed for missing endpoints.
`n.adapt`	the number of iterations for adaptation in the Markov chain Monte Carlo (MCMC) algorithm. The default is 1,000. This argument and the following `n.chains`, `n.burnin`, `n.iter`, and `n.thin` are passed to the functions in the package **rjags**.
`n.chains`	the number of MCMC chains. The default is 3.
`n.burnin`	the number of iterations for burn-in period. The default is 10,000.
`n.iter`	the total number of iterations in each MCMC chain after the burn-in period. The default is 10,000.
`n.thin`	a positive integer indicating thinning rate. The default is 1.
`data.name`	a character string indicating the data name. This is used in the names of the generated files that contain results. The default is `NULL`.
`traceplot`	a logical value indicating whether to save trace plots for the overall effect sizes and between-study standard deviations. The default is `FALSE`.
`coda`	a logical value indicating whether to output MCMC posterior samples. The default is `FALSE`.

Details

Suppose n studies are collected in a multivariate meta-analysis on a total of p endpoints. Denote the p-dimensional vector of effect sizes as $\boldsymbol{y}_i$, and their within-study variances form a diagonal matrix $\mathbf{D}_i$. However, the within-study correlations are unknown. Then, the random-effects hybrid model is as follows (Riley et al., 2008; Lin and Chu, 2018):

$$\boldsymbol{y}_i \sim N(\boldsymbol{\mu}, (\mathbf{D}_i + \mathbf{T})^{1/2}\mathbf{R}(\mathbf{D}_i + \mathbf{T})^{1/2}),$$

where $\boldsymbol{\mu}$ represents the overall effect sizes across studies, $\mathbf{T} = diag(\tau_1^2, \ldots, \tau_p^2)$ consists of the between-study variances, and $\mathbf{R}$ is the marginal correlation matrix. Although the within-study correlations are unknown, this model accounts for both within- and between-study correlations by using the marginal correlation matrix.

Uniform priors $U(0, 10)$ are specified for the between-study standard deviations τ_j $(j = 1, \ldots, p)$. The correlation matrix can be written as $\mathbf{R} = \mathbf{L}\mathbf{L}^T$, where $\mathbf{L} = (L_{ij})$ is a lower triangular matrix with nonnegative diagonal elements. Also, $L_{11} = 1$ and for $i = 2, \ldots, p$, $L_{ij} = \cos\theta_{i2}$ if $j = 1$; $L_{ij} = (\prod_{k=2}^{j} \sin\theta_{ik})\cos\theta_{i,j+1}$ if $j = 2, \ldots, i-1$; and $L_{ij} = \prod_{k=2}^{i} \sin\theta_{ik}$ if $j = i$ (Lu and Ades, 2009; Wei and Higgins, 2013). Here, θ_{ij}'s are angle parameters for $2 \le j \le i \le p$, and $\theta_{ij} \in (0, \pi)$. Uniform priors are specified for the angle parameters: $\theta_{ij} \sim U(0, \pi)$.

Value

This functions produces posterior estimates and Gelman and Rubin's potential scale reduction factor, and it generates several files that contain trace plots (if `traceplot = TRUE`), and MCMC posterior samples (if `coda = TRUE`) in users' working directory. In these results, `mu` represents the overall effect sizes, `tau` represents the between-study variances, `R` contains the elements of the correlation matrix, and `theta` represents the angle parameters (see "Details").

References

Lin L, Chu H (2018), "Bayesian multivariate meta-analysis of multiple factors." *Research Synthesis Methods*, **9**(2), 261–272. <doi: 10.1002/jrsm.1293>

Lu G, Ades AE (2009). "Modeling between-trial variance structure in mixed treatment comparisons." *Biostatistics*, **10**(4), 792–805. <doi: 10.1093/biostatistics/kxp032>

Riley RD, Thompson JR, Abrams KR (2008), "An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown." *Biostatistics*, **9**(1), 172–186. <doi: 10.1093/biostatistics/kxm023>

Wei Y, Higgins JPT (2013). "Bayesian multivariate meta-analysis with multiple outcomes." *Statistics in Medicine*, **32**(17), 2911–2934. <doi: 10.1002/sim.5745>

See Also

`mvma`, `mvma.bayesian`, `mvma.hybrid`

Examples

```
data("dat.pte")
set.seed(12345)
## increase n.burnin and n.iter for better convergence of MCMC
out <- mvma.hybrid.bayesian(ys = dat.pte$y, vars = (dat.pte$se)^2,
  n.adapt = 1000, n.chains = 3, n.burnin = 100, n.iter = 100,
  n.thin = 1, data.name = "Pterygium")
out
```

`pb.bayesian.binary` *Bayesian Method for Assessing Publication Bias in Meta-Analysis of a Binary Outcome*

Description

Performs multiple methods introduced in Shi et al. (2020) to assess publication bias under the Bayesian framework in a meta-analysis of (log) odds ratios.

Usage

```
pb.bayesian.binary(n00, n01, n10, n11, p01 = NULL, p11 = NULL, data,
                   sig.level = 0.1,  method = "bay", het = "mul",
                   sd.prior = "unif", n.adapt = 1000, n.chains = 3,
                   n.burnin = 5000, n.iter = 10000, thin = 2,
                   upp.het = 2, phi = 0.5, coda = FALSE,
                   traceplot = FALSE, seed = 1234)
```

Arguments

`n00`	a numeric vector or the corresponding column name in the argument `data`, indicating the counts of non-events in treatment group 0 in the collected studies.
`n01`	a numeric vector or the corresponding column name in the argument `data`, indicating the counts of events in treatment group 0 in the collected studies.
`n10`	a numeric vector or the corresponding column name in the argument `data`, indicating the counts of non-events in treatment group 1 in the collected studies.
`n11`	a numeric vector or the corresponding column name in the argument `data`, indicating the counts of events in treatment group 1 in the collected studies.
`p01`	an optional numeric vector indicating true event rates (e.g., from simulations) in the treatment group 0 across studies.
`p11`	an optional numeric vector indicating true event rates (e.g., from simulations) in the treatment group 1 across studies.
`data`	an optional data frame containing the meta-analysis dataset. If `data` is specified, the previous arguments, `n00`, `n01`, `n10`, `n11`, `p01` (if any), and `p11` (if any) should be specified as their corresponding column names in `data`.
`sig.level`	a numeric value indicating the statistical significance level α for testing for publication bias. The default is 0.1. It corresponds to $(1-\alpha)\times 100\%$ confidence/credible intervals.
`method`	a character string specifying the method for assessing publication bias via Bayesian hierarchical models. It can be one of `"bay"` (the Bayesian approach proposed in Shi et al., 2020), `"reg.bay"` (Egger's regression test, see Egger et al., 1997, under the Bayesian framework) and `"smoothed.bay"` (the regression test based on the smoothed sample variance, see Jin et al., 2014, under the Bayesian framework), where all regression tests are under the random-effects setting. The default is `"bay"`.
`het`	a character string specifying the type of heterogeneity assumption for the publication bias tests. It can be either `"mul"` (multiplicative heterogeneity assumption; see Thompson and Sharpe, 1999) or `"add"` (additive heterogeneity assumption). The default is `"mul"`.
`sd.prior`	a character string specifying prior distributions for standard deviation parameters. It can be either `"unif"` (uniform distribution) or `"hn"` (half-normal distribution). The default is `"unif"`.
`n.adapt`	the number of iterations for adaptation in the Markov chain Monte Carlo (MCMC) algorithm. The default is 1,000. This argument and the following `n.chains`, `n.burnin`, `n.iter`, and `thin` are passed to the functions in the package **rjags**.

`n.chains`	the number of MCMC chains. The default is 1.
`n.burnin`	the number of iterations for burn-in period. The default is 5,000.
`n.iter`	the total number of iterations in each MCMC chain after the burn-in period. The default is 10,000.
`thin`	a positive integer indicating thinning rate. The default is 2.
`upp.het`	a positive number for specifying the upper bound of uniform priors for standard deviation parameters (if `sd.prior = "unif"`). The default is 2.
`phi`	a positive number for specifying the hyper-parameter of half-normal priors for standard deviation parameters (if `sd.prior = "hn"`). The default is 0.5.
`coda`	a logical value indicating whether to output MCMC posterior samples. The default is `FALSE`.
`traceplot`	a logical value indicating whether to draw trace plots for the regression slopes. The default is `FALSE`.
`seed`	an integer for specifying the seed value for reproducibility.

Details

The Bayesian models are specified in Shi et al. (2020). The vague prior N(0, 10^4) is used for the regression intercept and slope, and the uniform prior U(0, `upp.het`) and half-normal prior HN(`phi`) are used for standard deviation parameters. The half-normal priors may be preferred in meta-analyses with rare events or small sample sizes.

Value

This function returns a list containing estimates of regression slopes and their credible intervals with the specified significance level (`sig.level`) as well as MCMC posterior samples (if `coda = TRUE`). Each element name in this list is related to a certain publication bias method (e.g., `est.bay` and `ci.bay` represent the slope estimate and its credible interval based on the proposed Bayesian method). In addition, traceplots for the regression slope are drawn if `traceplot = TRUE`.

Note

The current version does not support other effect measures such as relative risks or risk differences.

Author(s)

Linyu Shi <ls16d@my.fsu.edu>

References

Egger M, Davey Smith G, Schneider M, Minder C (1997). "Bias in meta-analysis detected by a simple, graphical test." *BMJ*, **315**(7109), 629–634. <doi: 10.1136/bmj.315.7109.629>

Jin Z-C, Wu C, Zhou X-H, He J (2014). "A modified regression method to test publication bias in meta-analyses with binary outcomes." *BMC Medical Research Methodology*, **14**, 132. <doi: 10.1186/1471228814132>

Shi L, Chu H, Lin L (2020). "A Bayesian approach to assessing small-study effects in meta-analysis of a binary outcome with controlled false positive rate". *Research Synthesis Methods*, **11**(4), 535–552. <doi: 10.1002/jrsm.1415>

Thompson SG, Sharp SJ (1999). "Explaining heterogeneity in meta-analysis: a comparison of methods." *Statistics in Medicine*, **18**(20), 2693–2708. <doi: 10.1002/(SICI)10970258(19991030)18:20<2693::AID-SIM235>3.0.CO;2V>

See Also

`pb.hybrid.binary`, `pb.hybrid.generic`

Examples

```
data("dat.poole")
set.seed(654321)
## increase n.burnin and n.iter for better convergence of MCMC
rslt.poole <- pb.bayesian.binary(n00, n01, n10, n11, data = dat.poole,
  method = "bay", het = "mul", sd.prior = "unif", n.adapt = 1000,
  n.chains = 3, n.burnin = 500, n.iter = 2000, thin = 2, upp.het = 2)
rslt.poole

data("dat.ducharme")
set.seed(654321)
## increase n.burnin and n.iter for better convergence of MCMC
rslt.ducharme <- pb.bayesian.binary(n00, n01, n10, n11, data = dat.ducharme,
  method = "bay", het = "mul", sd.prior = "unif", n.adapt = 1000,
  n.chains = 3, n.burnin = 500, n.iter = 2000, thin = 2, upp.het = 2)
rslt.ducharme

data("dat.henry")
set.seed(654321)
## increase n.burnin and n.iter for better convergence of MCMC
rslt.henry <- pb.bayesian.binary(n00, n01, n10, n11, data = dat.henry,
  method = "bay", het = "mul", sd.prior = "unif", n.adapt = 1000,
  n.chains = 3, n.burnin = 500, n.iter = 2000, thin = 2, upp.het = 2)
rslt.henry
```

`pb.hybrid.binary` *Hybrid Test for Publication Bias in Meta-Analysis With Binary Outcomes*

Description

Performs the hybrid test for publication bias introduced in Lin (2020), which synthesizes results from multiple popular publication bias tests, in a meta-analysis with binary outcomes.

Usage

```
pb.hybrid.binary(n00, n01, n10, n11, data, methods,
                 iter.resam = 1000, theo.pval = TRUE)
```

Arguments

`n00` a numeric vector or the corresponding column name in the argument `data`, indicating the counts of non-events in treatment group 0 in the collected studies.

`n01` a numeric vector or the corresponding column name in the argument `data`, indicating the counts of events in treatment group 0 in the collected studies.

`n10` a numeric vector or the corresponding column name in the argument `data`, indicating the counts of non-events in treatment group 1 in the collected studies.

`n11` a numeric vector or the corresponding column name in the argument `data`, indicating the counts of events in treatment group 1 in the collected studies.

`data` an optional data frame containing the meta-analysis dataset. If `data` is specified, the previous arguments, `n00`, `n01`, `n10`, and `n11`, should be specified as their corresponding column names in `data`.

`methods` a vector of character strings indicating the publication bias tests to be included in the hybrid test. They can be a subset of `"rank"` (Begg's rank test; see Begg and Mazumdar, 1994), `"reg"` (Egger's regression test under the fixed-effect setting; see Egger et al., 1997), `"reg.het"` (Egger's regression test accounting for additive heterogeneity), `"skew"` (the skewness-based test under the fixed-effect setting; see Lin and Chu, 2018), `"skew.het"` (the skewness-based test accounting for additive heterogeneity), `"inv.sqrt.n"` (the regression test based on sample sizes; see Tang and Liu, 2000), `"trimfill"` (the trim-and-fill method; see Duval and Tweedie, 2000), `"n"` (the regressoin test with sample sizes as the predictor; see Macaskill et al., 2001), `"inv.n"` (the regressoin test with the inverse of sample sizes as the predictor; see Peters et al., 2006), `"as.rank"` (the rank test based on the arcsine-transformed effect sizes; see Rucker et al., 2008), `"as.reg"` (the regression test based on the arcsine-transformed effect sizes under the fixed-effect setting), `"as.reg.het"` (the regression test based on the arcsine-transformed effect sizes accounting for additive heterogeneity), `"smoothed"` (the regression test based on the smoothed sample variances under the fixed-effect setting; see Jin et al., 2014), `"smoothed.het"` (the regression test based on the smoothed sample variances accounting for additive heterogeneity), `"score"` (the regression test based on the score function; see Harbord et al., 2006), and `"count"` (the test based on the hypergeometric distributions of event counts, designed for sparse data; see Schwarzer et al., 2007). The default is to include all aforementioned tests.

`iter.resam` a positive integer indicating the number of resampling iterations for calculating the p-value of the hybrid test.

`theo.pval` a logical value indicating whether additionally calculating the p-values of the tests specified in `methods` based on the test statistics' theoretical null distributions. Regardless of this argument, the resampling-based p-values are always produced by this function for the tests specified in `methods`.

Details

The hybrid test statistic is defined as the minimum p-value among the publication bias tests considered in the set specified by the argument `methods`. Note that the minimum p-value is no longer a genuine p-value because it cannot control the type I error rate. Its p-value needs to be calculated via the resampling approach. See more details in Lin (2020).

Value

This function returns a list containing p-values of the publication bias tests specified in `methods` as well as the hybrid test. Each element's name in this list has the format of `pval.x`, where `x` stands for the character string corresponding to a certain publication bias test, such as `rank`, `reg`, `skew`, etc. The hybrid test's p-value has the name `pval.hybrid`. If `theo.pval` = `TRUE`, additional elements of p-values of the tests in `methods` based on theorectical null distributions are included in the produced list; their names have the format of `pval.x.theo`. Another p-value of the hybrid test is also produced based on them; its corresponding element has the name `pval.hybrid.theo`.

References

Begg CB, Mazumdar M (1994). "Operating characteristics of a rank correlation test for publication bias." *Biometrics*, **50**(4), 1088–1101. <doi: 10.2307/2533446>

Duval S, Tweedie R (2000). "A nonparametric 'trim and fill' method of accounting for publication bias in meta-analysis." *Journal of the American Statistical Association*, **95**(449), 89–98. <doi: 10.1080/01621459.2000.10473905>

Egger M, Davey Smith G, Schneider M, Minder C (1997). "Bias in meta-analysis detected by a simple, graphical test." *BMJ*, **315**(7109), 629–634. <doi: 10.1136/bmj.315.7109.629>

Harbord RM, Egger M, Sterne JAC (2006). "A modified test for small-study effects in meta-analyses of controlled trials with binary endpoints." *Statistics in Medicine*, **25**(20), 3443–3457. <doi: 10.1002/sim.2380>

Jin Z-C, Wu C, Zhou X-H, He J (2014). "A modified regression method to test publication bias in meta-analyses with binary outcomes." *BMC Medical Research Methodology*, **14**, 132. <doi: 10.1186/1471228814132>

Lin L (2020). "Hybrid test for publication bias in meta-analysis." *Statistical Methods in Medical Research*, in press. <doi: 10.1177/0962280220910172>

Lin L, Chu H (2018). "Quantifying publication bias in meta-analysis." *Biometrics*, **74**(3), 785–794. <doi: 10.1111/biom.12817>

Macaskill P, Walter SD, Irwig L (2001). "A comparison of methods to detect publication bias in meta-analysis." *Statistics in Medicine*, **20**(4), 641–654. <doi: 10.1002/sim.698>

Peters JL, Sutton AJ, Jones DR, Abrams KR, Rushton L (2006). "Comparison of two methods to detect publication bias in meta-analysis." *JAMA*, **295**(6), 676–680. <doi: 10.1001/jama.295.6.676>

Rucker G, Schwarzer G, Carpenter J (2008). "Arcsine test for publication bias in meta-analyses with binary outcomes." *Statistics in Medicine*, **27**(5), 746–763. <doi: 10.1002/sim.2971>

Schwarzer G, Antes G, Schumacher M (2007). "A test for publication bias in meta-analysis with sparse binary data." *Statistics in Medicine*, **26**(4), 721–733. <doi: 10.1002/sim.2588>

Tang J-L, Liu JLY (2000). "Misleading funnel plot for detection of bias in meta-analysis." *Journal of Clinical Epidemiology*, **53**(5), 477–484. <doi: 10.1016/S08954356(99)002048>

Thompson SG, Sharp SJ (1999). "Explaining heterogeneity in meta-analysis: a comparison of methods." *Statistics in Medicine*, **18**(20), 2693–2708. <doi: 10.1002/(SICI)10970258(19991030)18:20<2693::AID-SIM235>3.0.CO;2V>

See Also

pb.bayesian.binary, pb.hybrid.generic

Examples

```
## meta-analysis of (log) odds ratios
data("dat.whiting")
# based on only 10 resampling iterations
set.seed(1234)
out.whiting <- pb.hybrid.binary(n00 = n00, n01 = n01,
  n10 = n10, n11 = n11, data = dat.whiting, iter.resam = 10)
out.whiting
# increases the number of resampling iterations to 10000,
#  taking longer time
```

---

pb.hybrid.generic — *Hybrid Test for Publication Bias in Meta-Analysis With Generic Outcomes*

Description

Performs the hybrid test for publication bias introduced in Lin (2020), which synthesizes results from multiple popular publication bias tests, in a meta-analysis with generic outcomes.

Usage

```
pb.hybrid.generic(y, s2, n, data, methods,
                    iter.resam = 1000, theo.pval = TRUE)
```

Arguments

`y`	a numeric vector or the corresponding column name in the argument `data`, indicating the observed effect sizes in the collected studies.
`s2`	a numeric vector or the corresponding column name in the argument `data`, indicating the within-study variances.
`n`	an optional numeric vector or the corresponding column name in the argument `data`, indicating the study-specific total sample sizes. This argument is required if the sample-size-based test (`"inv.sqrt.n"`) is included in `method`.
`data`	an optional data frame containing the meta-analysis dataset. If `data` is specified, the previous arguments, `y`, `s2`, and `n`, should be specified as their corresponding column names in `data`.

`methods` a vector of character strings indicating the publication bias tests to be included in the hybrid test. They can be a subset of `"rank"` (Begg's rank test; see Begg and Mazumdar, 1994), `"reg"` (Egger's regression test under the fixed-effect setting; see Egger et al., 1997), `"reg.het"` (Egger's regression test accounting for additive heterogeneity), `"skew"` (the skewness-based test under the fixed-effect setting; see Lin and Chu, 2018), `"skew.het"` (the skewness-based test accounting for additive heterogeneity), `"inv.sqrt.n"` (the regression test based on sample sizes; see Tang and Liu, 2000), and `"trimfill"` (the trim-and-fill method; see Duval and Tweedie, 2000). The default is to include all aforementioned tests.

`iter.resam` a positive integer indicating the number of resampling iterations for calculating the p-value of the hybrid test.

`theo.pval` a logical value indicating whether additionally calculating the p-values of the tests specified in `methods` based on the test statistics' theoretical null distributions. Regardless of this argument, the resampling-based p-values are always produced by this function for the tests specified in `methods`.

Details

The hybrid test statistic is defined as the minimum p-value among the publication bias tests considered in the set specified by the argument `methods`. Note that the minimum p-value is no longer a genuine p-value because it cannot control the type I error rate. Its p-value needs to be calculated via the resampling approach. See more details in Lin (2020).

Value

This function returns a list containing p-values of the publication bias tests specified in `methods` as well as the hybrid test. Each element's name in this list has the format of `pval.x`, where `x` stands for the character string corresponding to a certain publication bias test, such as `rank`, `reg`, `skew`, etc. The hybrid test's p-value has the name `pval.hybrid`. If `theo.pval` = `TRUE`, additional elements of p-values of the tests in `methods` based on theorectical null distributions are included in the produced list; their names have the format of `pval.x.theo`. Another p-value of the hybrid test is also produced based on them; its corresponding element has the name `pval.hybrid.theo`.

References

Begg CB, Mazumdar M (1994). "Operating characteristics of a rank correlation test for publication bias." *Biometrics*, **50**(4), 1088–1101. <doi: 10.2307/2533446>

Duval S, Tweedie R (2000). "A nonparametric 'trim and fill' method of accounting for publication bias in meta-analysis." *Journal of the American Statistical Association*, **95**(449), 89–98. <doi: 10.1080/01621459.2000.10473905>

Egger M, Davey Smith G, Schneider M, Minder C (1997). "Bias in meta-analysis detected by a simple, graphical test." *BMJ*, **315**(7109), 629–634. <doi: 10.1136/bmj.315.7109.629>

Lin L (2020). "Hybrid test for publication bias in meta-analysis." *Statistical Methods in Medical Research*, in press. <doi: 10.1177/0962280220910172>

Lin L, Chu H (2018). "Quantifying publication bias in meta-analysis." *Biometrics*, **74**(3), 785–794. <doi: 10.1111/biom.12817>

Tang J-L, Liu JLY (2000). "Misleading funnel plot for detection of bias in meta-analysis." *Journal of Clinical Epidemiology*, **53**(5), 477–484. <doi: 10.1016/S08954356(99)002048>

Thompson SG, Sharp SJ (1999). "Explaining heterogeneity in meta-analysis: a comparison of methods." *Statistics in Medicine*, **18**(20), 2693–2708. <doi: 10.1002/(SICI)10970258(19991030)18:20<2693::AID-SIM235>3.0.CO;2V>

See Also

`pb.bayesian.binary`, `pb.hybrid.binary`

Examples

```
## meta-analysis of mean differences
data("dat.plourde")
# based on only 10 resampling iterations
set.seed(1234)
out.plourde <- pb.hybrid.generic(y = y, s2 = s2, n = n,
  data = dat.plourde, iter.resam = 10)
out.plourde
# only produces resampling-based p-values
set.seed(1234)
pb.hybrid.generic(y = y, s2 = s2, n = n,
  data = dat.plourde, iter.resam = 10, theo.pval = FALSE)
# increases the number of resampling iterations to 10000,
#  taking longer time

## meta-analysis of standardized mean differences
data("dat.paige")
# based on only 10 resampling iterations
set.seed(1234)
out.paige <- pb.hybrid.generic(y = y, s2 = s2, n = n,
  data = dat.paige, iter.resam = 10)
out.paige
# increases the number of resampling iterations to 10000,
#  taking longer time
```

`plot.metaoutliers` *Standardized Residual Plot for Outliers Diagnostics*

Description

Draws a plot showing study-specific standardized residuals.

Usage

```
## S3 method for class 'metaoutliers'
plot(x, xtick.cex = 1, ytick.cex = 0.5, ...)
```

Arguments

`x`	an object created by the function `metaoutliers`.
`xtick.cex`	a numeric value indicating the magnification to be used for ticks on the x axis.
`ytick.cex`	a numeric value indicating the magnification to be used for ticks on the y axis.
`...`	Other arguments that can be passed to the function `plot.default`.

Value

None.

See Also

`metaoutliers`

Examples

```
data("dat.aex")
attach(dat.aex)
out.aex <- metaoutliers(y, s2, model = "FE")
detach(dat.aex)
plot(out.aex)

data("dat.hipfrac")
attach(dat.hipfrac)
out.hipfrac <- metaoutliers(y, s2, model = "RE")
detach(dat.hipfrac)
plot(out.hipfrac)
```

---

ssfunnel *Contour-Enhanced Sample-Size-Based Funnel Plot*

Description

Generates contour-enhanced sample-size-based funnel plot for a meta-analysis of mean differences, standardized mean differences, (log) odds ratios, (log) relative risks, or risk differences.

Usage

```
ssfunnel(y, s2, n, data, type, alpha = c(0.1, 0.05, 0.01, 0.001),
         log.ss = FALSE, sigma, p0, xlim, ylim, xlab, ylab,
         cols.contour, col.mostsig, cex.pts, lwd.contour, pch,
         x.legend, y.legend, cex.legend, bg.legend, ...)
```

Arguments

y	a numeric vector or the corresponding column name in the argument data, indicating the observed effect sizes in the collected studies.
s2	a numeric vector or the corresponding column name in the argument data, indicating the within-study variances.
n	a numeric vector or the corresponding column name in the argument data, indicating the study-specific total sample sizes.
data	an optional data frame containing the meta-analysis dataset. If data is specified, the previous arguments, y, s2, and n, should be specified as their corresponding column names in data.
type	a character string indicating the type of effect size, which should be one of "md" (mean difference), "smd" (standardized mean difference), "lor" (log odds ratio), "lrr" (log relative risk), and "rd" (risk difference).
alpha	a numeric vector indicating the significance levels to be presented in the sample-size-based funnel plot.
log.ss	a logical value indicating whether sample sizes are plotted on a logarithmic scale (TRUE) or not (FALSE, the default).
sigma	a positive numeric value that is required for the mean difference (type = "md"), indicating a rough estimate of the common standard deviation of the samples' continuous outcomes in the two groups across studies. It is not used for other effect size types.
p0	an optional numeric value indicating a rough estimate of the common event rate in the control group across studies. It is only used for the (log) odds ratio, (log) relative risk, and risk difference.
xlim	the x limits c(x1,x2) of the plot.
ylim	the y limits c(y1,y2) of the plot.
xlab	a label for the x axis.
ylab	a label for the y axis.
cols.contour	a vector of character strings; they indicate colors of the contours to be presented in the sample-size-based funnel plot, and correspond to the significance levels specified in the argument alpha.
col.mostsig	a character string indicating the color for the most significant result among the studies in the meta-analysis.
cex.pts	the size of the points.
lwd.contour	the width of the contours.
pch	the symbol of the points.
x.legend	the x co-ordinate or a keyword, such as "topleft" (the default), to be used to position the legend. It is passed to legend.
y.legend	the y co-ordinate to be used to position the legend (the default is NULL).
cex.legend	the size of legend text.
bg.legend	the background color for the legend box.
...	other arguments that can be passed to plot.default.

Details

A contour-enhanced sample-size-based funnel plot is generated; it presents study-specific total sample sizes against the corresponding effect size estimates. It is helpful to avoid the confounding effect caused by the intrinsic association between effect size estimates and standard errors in the conventional standard-error-based funnel plot. See details of the derivations of the contours in Lin (2019).

Value

None.

References

Lin L (2019). "Graphical augmentations to sample-size-based funnel plot in meta-analysis." *Research Synthesis Methods*, **10**(3), 376–388. <doi: 10.1002/jrsm.1340>

Peters JL, Sutton AJ, Jones DR, Abrams KR, Rushton L (2006). "Comparison of two methods to detect publication bias in meta-analysis." *JAMA*, **295**(6), 676–680. <doi: 10.1001/jama.295.6.676>

Examples

```
## mean difference
data("dat.annane")
# descriptive statistics for sigma (continuous outcomes' standard deviation)
quantile(sqrt(dat.annane$s2/(1/dat.annane$n1 + 1/dat.annane$n2)),
  probs = c(0, 0.25, 0.5, 0.75, 1))
# based on sigma = 8
ssfunnel(y, s2, n, data = dat.annane, type = "md",
  alpha = c(0.1, 0.05, 0.01, 0.001), sigma = 8)
# sample sizes presented on a logarithmic scale with plot title
ssfunnel(y, s2, n, data = dat.annane, type = "md",
  alpha = c(0.1, 0.05, 0.01, 0.001), sigma = 8, log.ss = TRUE,
  main = "Contour-enhanced sample-size-based funnel plot")
# based on sigma = 17, with specified x and y limits
ssfunnel(y, s2, n, data = dat.annane, type = "md",
  xlim = c(-15, 15), ylim = c(30, 500),
  alpha = c(0.1, 0.05, 0.01, 0.001), sigma = 17, log.ss = TRUE)
# based on sigma = 20
ssfunnel(y, s2, n, data = dat.annane, type = "md",
  xlim = c(-15, 15), ylim = c(30, 500),
  alpha = c(0.1, 0.05, 0.01, 0.001), sigma = 20, log.ss = TRUE)

## standardized mean difference
data("dat.barlow")
ssfunnel(y, s2, n, data = dat.barlow, type = "smd",
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-1.5, 1))

## log odds ratio
data("dat.butters")
ssfunnel(y, s2, n, data = dat.butters, type = "lor",
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-1.5, 1.5))
# use different colors for contours
ssfunnel(y, s2, n, data = dat.butters, type = "lor",
```

```
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-1.5, 1.5),
  cols.contour = c("blue", "green", "yellow", "red"), col.mostsig = "black")
# based on p0 = 0.3 (common event rate in the control group across studies)
ssfunnel(y, s2, n, data = dat.butters, type = "lor",
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-1.5, 1.5), p0 = 0.3)
# based on p0 = 0.5
ssfunnel(y, s2, n, data = dat.butters, type = "lor",
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-1.5, 1.5), p0 = 0.5)

## log relative risk
data("dat.williams")
ssfunnel(y, s2, n, data = dat.williams, type = "lrr",
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-1.5, 2.5))
# based on p0 = 0.2
ssfunnel(y, s2, n, data = dat.williams, type = "lrr",
  alpha = c(0.1, 0.05, 0.01, 0.001), p0 = 0.2, xlim = c(-1.5, 2.5))
# based on p0 = 0.3
ssfunnel(y, s2, n, data = dat.williams, type = "lrr",
  alpha = c(0.1, 0.05, 0.01, 0.001), p0 = 0.3, xlim = c(-1.5, 2.5))

## risk difference
data("dat.kaner")
ssfunnel(y, s2, n, data = dat.kaner, type = "rd",
  alpha = c(0.1, 0.05, 0.01, 0.001), xlim = c(-0.5, 0.5))
# based on p0 = 0.1
ssfunnel(y, s2, n, data = dat.kaner, type = "rd",
  alpha = c(0.1, 0.05, 0.01, 0.001), p0 = 0.1, xlim = c(-0.5, 0.5))
# based on p0 = 0.4
ssfunnel(y, s2, n, data = dat.kaner, type = "rd",
  alpha = c(0.1, 0.05, 0.01, 0.001), p0 = 0.4, xlim = c(-0.5, 0.5))
```

Index