Package 'metamisc'

May 13, 2018

Type Package

Title Diagnostic and Prognostic Meta-Analysis

Version 0.1.9

Date 2018-05-13

Description
Meta-analysis of diagnostic and prognostic modeling studies. Summarize estimates of prognostic factors, diagnostic test accuracy and prediction model performance. Validate, update and combine published prediction models. Develop new prediction models with data from multiple studies.

Imports metafor (>= 2.0.0), mvtnorm, ellipse, lme4, plyr, ggplot2

Depends R (>= 3.2.0), stats, graphics

Suggests runjags, rjags, testthat (>= 1.0.2)

License GPL-3

URL http://r-forge.r-project.org/projects/metamisc/

RoxygenNote 6.0.1

Author Thomas Debray [aut, cre] (<https://orcid.org/0000-0002-1790-2719>),
Valentijn de Jong [aut]

Maintainer Thomas Debray <thomas.debray@gmail.com>

Repository CRAN

Repository/R-Forge/Project metamisc

Repository/R-Forge/Revision 442

Repository/R-Forge/DateTimeStamp 2018-05-13 12:13:48

Date/Publication 2018-05-13 14:25:21 UTC

NeedsCompilation no

R topics documented:

`metamisc-package` *Diagnostic and Prognostic Meta-Analysis*

Description

Meta-analysis of diagnostic and prognostic modeling studies. Summarize estimates of prognostic factors, diagnostic test accuracy and prediction model performance. Validate, update and combine published prediction models.

Details

The following functionality is currently implemented: univariate meta-analysis of summary data (uvmeta), bivariate meta-analysis of correlated outcomes (riley), meta-analysis of prediction model performance (valmeta), evaluation of funnel plot asymmetry (fat).

The metamisc package also provides a comprehensive framework for developing prediction models when patient-level data from multiple studies or settings are available (metapred).

Author(s)

Thomas Debray <thomas.debray@gmail.com>, Valentijn de Jong

References

Debray TPA, Moons KGM, Ahmed I, Koffijberg H, Riley RD. A framework for developing, implementing, and evaluating clinical prediction models in an individual participant data meta-analysis. Stat Med. 2013;32(18):3158–80.

Debray TPA, Damen JAAG, Snell KIE, Ensor J, Hooft L, Reitsma JB, et al. A guide to systematic review and meta-analysis of prediction model performance. BMJ. 2017;356:i6460.

Debray TPA, Moons KGM, Riley RD. Detecting small-study effects and funnel plot asymmetry in meta-analysis of survival data: a comparison of new and existing tests. Res Syn Meth. 2018;9(1):41–50.

Debray TPA, Damen JAAG, Riley R, Snell KIE, Reitsma JB, Hooft L, et al. A framework for meta-analysis of prediction model studies with binary and time-to-event outcomes. Stat Methods Med Res. 2018; In press.

Riley RD, Thompson JR, Abrams KR. An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown. Biostatistics 2008; 9: 172–186.

See Also

fat, metapred, riley, uvmeta, valmeta

---

ccalc *Calculate the concordance statistic*

Description

The function calculates (transformed versions of) the concordance (c-) statistic with the corresponding sampling variance.

Usage

```
ccalc(cstat, cstat.se, cstat.cilb, cstat.ciub, cstat.cilv, sd.LP, N, O, Po,
  data, slab, subset, g = NULL, level = 0.95, approx.se.method = 4, ...)
```

Arguments

`cstat`	vector to specify the estimated c-statistics.
`cstat.se`	vector to specify the corresponding standard errors.
`cstat.cilb`	vector to specify the lower limits of the confidence interval.
`cstat.ciub`	vector to specify the upper limits of the confidence interval.
`cstat.cilv`	vector to specify the levels of aformentioned confidence interval limits. (default: 0.95, which corresponds to the 95% confidence interval).
`sd.LP`	vector to specify the standard deviations of the linear predictor (prognostic index).
`N`	vector to specify the sample/group sizes.
`O`	vector to specify the total number of observed events.
`Po`	vector to specify the observed event probabilities.
`data`	optional data frame containing the variables given to the arguments above.
`slab`	optional vector with labels for the studies.
`subset`	optional vector indicating the subset of studies that should be used. This can be a logical vector or a numeric vector indicating the indices of the studies to include.
`g`	a quoted string that is the function to transform estimates of the c-statistic; see the details below.
`level`	level for confidence interval, default `0.95`.
`approx.se.method`	integer specifying which method should be used for estimating the standard error of the c-statistic (Newcombe, 2006). So far, only method `2` and method `4` (default) have been implemented.
`...`	Additional arguments.

Details

The c-statistic is a measure of discrimination, and indicates the ability of a prediction model to distinguish between patients developing and not developing the outcome. The c-statistic typically ranges from 0.5 (no discriminative ability) to 1 (perfect discriminative ability).

By default, the function `ccalc` will derive the c-statistic of each study, together with the corresponding standard error and 95% confidence interval. However, it is also possible to calculate transformed versions of the c-statistic. Appropriate standard errors are then derived using the Delta method. For instance, the logit transformation can be applied by specifying `g="log(cstat/(1-cstat))"`.

Restoring the c-statistic: For studies where the c-statistic is missing, it is estimated from the standard deviation of the linear predictor (`theta.source="std.dev(LP)"`). The corresponding method is describec

Restoring the standard error of the c-statistic: When missing, the standard error of the c-statistic can be estimated from the confidence interval. Alternatively, the standard error can be approximated from a combination of the reported c-statistic, the total sample size and the total number of events (Newcombe, 2006). This can be achieved by adopting (a modification of) the method proposed by Hanley and McNeil, as specified in `approx.se.method`.

Value

An array with the following columns:

"theta" The (transformed) c-statistics.

"theta.se" Standard errors of the (transformed) c-statistics.

"theta.CIl" Lower confidence interval of the (transformed) c-statistics. The level is specified in `level`. Intervals are calculated on the same scale as `theta` by assuming a Normal distribution.

"theta.CIu" Upper confidence interval of the (transformed) c-statistics. The level is specified in `level`. Intervals are calculated on the same scale as `theta` by assuming a Normal distribution.

"theta.source" Method used for calculating the (transformed) c-statistic.

"theta.se.source" Method used for calculating the standard error of the (transformed) c-statistic.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

Debray TPA, Damen JAAG, Snell KIE, Ensor J, Hooft L, Reitsma JB, et al. A guide to systematic review and meta-analysis of prediction model performance. *BMJ*. 2017; 356:i6460.

Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*. 1982; 143(1):29–36.

Newcombe RG. Confidence intervals for an effect size measure based on the Mann-Whitney statistic. Part 2: asymptotic methods and evaluation. *Stat Med*. 2006; 25(4):559–73.

Snell KI, Ensor J, Debray TP, Moons KG, Riley RD. Meta-analysis of prediction model performance across multiple studies: Which scale helps ensure between-study normality for the C - statistic and calibration measures? *Statistical Methods in Medical Research*. 2017.

White IR, Rapsomaniki E, the Emerging Risk Factors Collaboration. Covariate-adjusted measures of discrimination for survival data. *Biom J*. 2015;57(4):592–613.

Examples

```
######### Validation of prediction models with a binary outcome #########
data(EuroSCORE)

# Calculate the c-statistic and its standard error
ccalc(cstat=c.index, cstat.se=se.c.index, cstat.cilb=c.index.95CIl, cstat.ciub=c.index.95CIu,
      N=n, O=n.events, data=EuroSCORE, slab=Study)

# Calculate the logit c-statistic and its standard error
ccalc(cstat=c.index, cstat.se=se.c.index, cstat.cilb=c.index.95CIl, cstat.ciub=c.index.95CIu,
      N=n, O=n.events, data=EuroSCORE, slab=Study, g="log(cstat/(1-cstat))")
```

`coef.metapred` *Extract the regression coefficients The* `coef` *function extracts the estimated model coefficients from objects of class* `"metapred"`.

Description

Extract the regression coefficients The `coef` function extracts the estimated model coefficients from objects of class `"metapred"`.

Usage

```
## S3 method for class 'metapred'
coef(object, ...)
```

Arguments

`object`	A fitted `metapred` object
`...`	Optional arguments (currently not supported).

`Collins` *Collins data*

Description

A meta-analysis of nine clinical trials investigating the effect of taking diuretics during pregnancy on the risk of pre-eclampsia.

Usage

```
data(Collins)
```

Format

A data frame with 9 observations on the following 2 variables.

`logOR` a numeric vector with treatment effect sizes (log odds ratio)

`SE` a numeric vector with the standard error of the treatment effect sizes

Source

Collins, R., Yusuf, S., Peto, R. Overview of randomised trials of diuretics in pregnancy. *British Medical Journal* 1985, **290**, 17–23.

Hardy, R.J. Thompson, S.G. A likelihood approach to meta-analysis with random effects. *Statistics in Medicine* 1996; **15**:619–629.

Examples

```
data(Collins)
```

Daniels *Daniels and Hughes data*

Description

Data frame with treatment differences in CD4 cell count.

Usage

```
data("Daniels")
```

Format

A data frame with 15 observations on the following 2 variables.

`Y1` Treatment differences for the log hazard ratio for the development of AIDS or death over 2 years.

`vars1` Error variances of `Y1`.

`Y2` Difference in mean change in CD4 cell count between baseline and 6 month for studies of the AIDS Clinical Trial Group

`vars2` Error variances of `Y2`.

Details

The `Daniels` data comprises 15 phase II/III randomized clinical trials of the HIV Disease Section of the Adult AIDS Clinical Trials Group of the National Institutes of Health, which had data available as of May 1996, which had at least six months of follow-up on some patients and in which at least one patient developed AIDS or died. The data were previously used by Daniels and Hughes (1997) to assess whether the change in CD4 cell count is a surrogate for time to either development of AIDS or death in drug trials of patients with HIV.

Source

Daniels MJ, Hughes MD. Meta-analysis for the evaluation of potential surrogate markers. *Statistics in Medicine* 1997; **16**: 1965–1982.

DVTipd *Hypothetical dataset for diagnosis of Deep Vein Thrombosis (DVT)*

Description

A hypothetical dataset with 500 subjects suspected of having deep vein thrombosis (DVT).

Usage

```
data(DVTipd)
```

Format

A data frame with 500 observations of 15 variables.

`sex` gender (0=female, 1=male)

`malign` active malignancy (0=no active malignancy, 1=active malignancy)

`par` paresis (0=no paresis, 1=paresis)

`surg` recent surgery or bedridden

`tend` tenderness venous system

`oachst` oral contraceptives or hst

`leg` entire leg swollen

`notraum` absence of leg trauma

`calfdif3` calf difference >= 3 cm

`pit` pitting edema

`vein` vein distension

`altdiagn` alternative diagnosis present

`histdvt` history of previous DVT

`ddimdich` dichotimized D-dimer value

`dvt` final diagnosis of DVT

Details

Hypothetical dataset derived from the Individual Participant Data Meta-Analysis from Geersing *et al* (2014). The dataset consists of consecutive outpatients with suspected deep vein thrombosis, with documented information on the presence or absence of proximal deep vein thrombosis (`dvt`) by an acceptable reference test. Acceptable such tests were either compression ultrasonography or venography at initial presentation, or, if venous imaging was not performed, an uneventful follow-up for at least three months.

Source

Geersing GJ, Zuithoff NPA, Kearon C, Anderson DR, Ten Cate-Hoek AJ, Elf JL, et al. Exclusion of deep vein thrombosis using the Wells rule in clinically important subgroups: individual patient data meta-analysis. *BMJ*. 2014;348:g1340.

Examples

```
data(DVTipd)
str(DVTipd)
summary(apply(DVTipd,2,as.factor))

## Develop a prediction model to predict presence of DVT
model.dvt <- glm("dvt~sex+oachst+malign+surg+notraum+vein+calfdif3+ddimdich",
                 family=binomial, data=DVTipd)
summary(model.dvt)
```

DVTmodels *Risk prediction models for diagnosing Deep Venous Thrombosis (DVT)*

Description

Previously published prediction models for predicting the presence of DVT.

Usage

```
data(DVTmodels)
```

Format

An object of the class `litmodels` with the following information for each literature model: the study-level descriptives ("`descriptives`"), the regression coefficient or weight for each predictor ("`weights`") and the error variance for each regression coefficient or weight ("`weights.var`").

Details

Previously, several models (Gagne, Oudega) and score charts (Wells, modified Wells, and Hamilton) have been published for evaluating the presence of DVT in suspected patients. These models combine information on mulitple predictors into a weighted sum, that can subsequently be used to obtain estimates of absolute risk. See `DVTipd` for more information on the predictors.

Source

Wells PS, Anderson DR, Bormanis J, Guy F, Mitchell M, Gray L, Clement C, Robinson KS, Lewandowski B. Value of assessment of pretest probability of deep-vein thrombosis in clinical management. *Lancet* 1997; **350**(9094):1795–1798. DOI: 10.1016/S0140-6736(97)08140-3.

Wells PS, Anderson DR, Rodger M, Forgie M, Kearon C, Dreyer J, Kovacs G, Mitchell M, Lewandowski B, Kovacs MJ. Evaluation of D-dimer in the diagnosis of suspected deep-vein thrombosis. *The New England Journal of Medicine* 2003; **349**(13):1227–1235. DOI: 10.1056/NEJMoa023153.

Gagne P, Simon L, Le Pape F, Bressollette L, Mottier D, Le Gal G. Clinical prediction rule for diagnosing deep vein thrombosis in primary care. *La Presse Medicale* 2009; **38**(4):525–533. DOI: 10.1016/j.lpm.2008.09.022.

Subramaniam RM, Snyder B, Heath R, Tawse F, Sleigh J. Diagnosis of lower limb deep venous thrombosis in emergency department patients: performance of Hamilton and modified Wells scores. *Annals of Emergency Medicine* 2006; **48**(6):678–685. DOI: 10.1016/j.annemergmed.2006.04.010.

Oudega R, Moons KGM, Hoes AW. Ruling out deep venous thrombosis in primary care. a simple diagnostic algorithm including D-dimer testing. *Thrombosis and Haemostasis* 2005; **94**(1):200–205. DOI: 10.1160/TH04-12-0829.

References

Debray TPA, Koffijberg H, Nieboer D, Vergouwe Y, Steyerberg EW, Moons KGM. Meta-analysis and aggregation of multiple published prediction models. *Stat Med*. 2014 Jun 30;33(14):2341–62.

Debray TPA, Koffijberg H, Vergouwe Y, Moons KGM, Steyerberg EW. Aggregating published prediction models with individual participant data: a comparison of different approaches. *Stat Med*. 2012 Oct 15;31(23):2697–712.

See Also

DVTipd

Examples

```
data(DVTmodels)
```

EuroSCORE *Predictive performance of EuroSCORE II*

Description

This data set contains estimates on the predictive performance of the European system for cardiac operative risk evaluation (EuroSCORE II) in patients undergoing cardiac surgery. Results are based on the original development study and 22 validations identified by Guida *et al*.

Usage

```
data("EuroSCORE")
```

Format

A data frame with 23 observations on the following 13 variables.

`Study` a vector with the first author of each validation study

`n` a numeric vector with the total number of patients on which performance estimates are based

`n.events` a numeric vector with the total number of observed events

`c.index` a numeric vector with the estimated concordance statistic of each validation

`se.c.index` a numeric vector with the standard error of the concordance statistics

`c.index.95CIl` a numeric vector with the lower bound of the 95% confidence interval of the estimated concordance statistics

`c.index.95CIu` a numeric vector with the upper bound of the 95% confidence interval of the estimated concordance statistics

`Po` a numeric vector with the overall observed event probability of each validation

`Pe` a numeric vector with the overall expected event probability of each validation

`SD.Pe` a numeric vector with the standard error of `Pe`

`e.events` a numeric vector with the total number of expected events in each validation

`multicentre` a logical vector describing whether the study was a multicentre study

`mean.age` a numeric vector describing the mean age of the patients

`sd.age` a numeric vector with the spread of the age of the patients

`pts.before.2010` a logical vector describing whether studies included patients before 2010 (i.e., before EuroSCORE II was developed)

Details

Published in 2012, EuroSCORE II was developed using logistic regression in a dataset comprising 16,828 adult patients undergoing major cardiac surgery from 154 hospitals in 43 countries over a 12-week period (May-July) in 2010. EuroSCORE II was developed to predict in-hospital mortality for patients undergoing any type of cardiac surgery. In 2014, a systematic review of published evidence on the performance value of the euroSCORE II was undertaken by Guida *et al*. Twenty-two validations, including more 145,592 patients from 21 external validation articles (one study included two validations) and a split-sample validation contained within original development article were included in the review; 23 validation studies in total.

Source

Guida P, Mastro F, Scrascia G, Whitlock R, Paparella D. Performance of the European System for Cardiac Operative Risk Evaluation II: a meta-analysis of 22 studies involving 145,592 cardiac surgery procedures. *J Thorac Cardiovasc Surg*. 2014; **148**(6):3049–3057.e1.

Nashef SAM, Roques F, Sharples LD, Nilsson J, Smith C, Goldstone AR, et al. EuroSCORE II. *Eur J Cardiothorac Surg*. 2012; **41**(4):734-744; discussion 744-745.

Examples

```
data(EuroSCORE)
```

---

fat *Regression tests for detecting funnel plot asymmetry*

Description

The presence of small-study effects is a common threat to systematic reviews and meta-analyses, especially when it is due to publication bias, which occurs when small primary studies are more likely to be reported (published) if their findings were positive. The presence of small-study effects can be verified by visual inspection of the funnel plot, where for each included study of the meta-analysis, the estimate of the reported effect size is depicted against a measure of precision or sample size. The premise is that the scatter of plots should reflect a funnel shape, if small-study effects do not exist. However, when small studies are predominately in one direction (usually the direction of larger effect sizes), asymmetry will ensue.

The `fat` function implements several tests for detecting funnel plot asymmetry, which can be used when the presence of between-study heterogeneity in treatment effect is relatively low.

Usage

```
fat(b, b.se, n.total, d.total, d1, d2, method = "E-FIV")
```

Arguments

b	Vector with the effect size of each study. Examples are log odds ratio, log hazards ratio, log relative risk.
b.se	Optional vector with the standard error of the effect size of each study
n.total	Optional vector with the total sample size of each study
d.total	Optional vector with the total number of observed events for each study
d1	Optional vector with the total number of observed events in the exposed groups
d2	Optional vector with the total number of observed events in the unexposed groups
method	Method for testing funnel plot asymmetry, defaults to "E-FIV" (Egger's test with multiplicative dispersion). Other options are E-UW, M-FIV, M-FPV, D-FIV and D-FAV. More info in "Details"

Details

Egger regression method: A common method to test the presence of small-study effects is given as the following unweighted regression model (method="E-UW", Egger 1997):

$$\hat{b}_k = \beta_0 + \beta_1 \widehat{\mathrm{SE}}(\hat{b}_k) + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}\left(0, \sigma^2\right)$$

Whereas β_0 indicates the size and direction of the treatment effect, β_1 provides a measure of asymmetry; the larger its deviation from zero the more pronounced the asymmetry. Otherwise, if $\beta_1 = 0$, there is no association between the estimated effect sizes $\hat{b}_k$ and their corresponding estimates for the standard error $\widehat{\mathrm{SE}}(\hat{b}_k)$ among the reported studies, indicating no asymmetry and thus no small-study effects.

It is possible to allow for between-study heterogeneity by adopting a multiplicative overdispersion parameter by which the variance in each study is multiplied (method="E-FIV", Sterne 2000):

$$\hat{\beta}_k = a + b\,\widehat{\mathrm{SE}}(\hat{\beta}_k) + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}(0, \phi\ \widehat{\mathrm{var}}(\hat{\beta}_k))$$

Unfortunately, both tests are known to be intrinsically biased because: (i) the independent variable is subject to sampling variability; (ii) the standardized treatment effect is correlated with its estimated precision; and (iii) for binary data, the independent regression variable is a biased estimate of the true precision, with larger bias for smaller sample sizes (Macaskill 2001).

Macaskill regression method: To overcome the problems with the Egger approach, Macaskill et al. consider fitting a regression directly to the data using the treatment effect as the dependent variable, and study size (n_k) as the independent variable. Again, the observations are weighted by the inverse variance of the estimate to allow for possible heteroscedasticity (method="M-FIV", Macaskill 2001):

$$\hat{\beta}_k = a + b\,n_k + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}(0, \phi\ \widehat{\mathrm{var}}(\hat{\beta}_k))$$

Macaskill et al. also proposed an alternative test where a 'pooled' estimate of the outcome proportion is used for the variance $\widehat{\text{var}}(\hat{b}_k)$ (`method="M-FPV"`, Macaskill 2001):

$$\hat{\beta}_k = a + b\, n_k + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}\left(0, \phi\ \frac{1}{d_k(1 - d_k/n_k)}\right)$$

For studies with zero events, a continuity correction is applied by adding 0.5 to all cell counts.

Peters regression method: A modification of Macaskill's test was proposed by Peters et al. to obtain more balanced type-I error rates in the tail probability areas (`method="P-FPV"`, Peters 2006):

$$\hat{\beta}_k = a + b\, \frac{1}{n_k} + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}\left(0, \phi\ \frac{1}{d_k(1 - d_k/n_k)}\right)$$

Again, 0.5 is added to all cells for studies with zero events.

Debray regression method: Because the use of aforementioned tests may be less appropriate in the presence of survival data, Debray et al. proposed using the total number of events (d_k) as independent variable (`method="D-FIV"`, Debray 2017):

$$\hat{\beta}_k = a + b\, \frac{1}{d_k} + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}(0, \phi\ \widehat{\text{var}}(\hat{\beta}_k))$$

For studies with zero events, the total number of observed events is set to 1. Alternatively, when $\widehat{\text{var}}(\hat{\beta}_k)$ is unknown or derived from small samples, Debray at al.proposed to use the following regression model (`method="D-FAV"`, Debray 2017):

$$\hat{\beta}_k = a + b\, \frac{1}{d_k} + \epsilon_k \ , \ \epsilon_k \sim \mathcal{N}\left(0, \phi\ \left(\frac{1}{d_{k1}} + \frac{1}{d_{k2}}\right)\right)$$

Value

a list containing the following entries:

"pval" A two-sided P-value indicating statistical significance of the funnel plot asymettry test. Values below the significance level (usually defined as 10%) support the presence of funnel plot asymmetry, and thus small-study effects.

"model" A fitted `glm` object, representing the estimated regression model used for testing funnel plot asymmetry.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

Debray TPA, Moons KGM, Riley RD. Detecting small-study effects and funnel plot asymmetry in meta-analysis of survival data: a comparison of new and existing tests. Res Syn Meth. 2018;9(1):41–50.

Egger M, Davey Smith G, Schneider M, Minder C. Bias in meta-analysis detected by a simple, graphical test. *BMJ*. 1997;315(7109):629–34.

Macaskill P, Walter SD, Irwig L. A comparison of methods to detect publication bias in meta-analysis. *Stat Med*. 2001;20(4):641–54.

Peters JL, Sutton AJ, Jones DR, Abrams KR, Rushton L. Comparison of two methods to detect publication bias in meta-analysis. *JAMA*. 2006 Feb 8;295(6):676–80.

Sterne JA, Gavaghan D, Egger M. Publication and related bias in meta-analysis: power of statistical tests and prevalence in the literature. *J Clin Epidemiol*. 2000;53(11):1119–29.

See Also

plot.fat

Examples

```
data(Fibrinogen)
b <- log(Fibrinogen$HR)
b.se <- ((log(Fibrinogen$HR.975) - log(Fibrinogen$HR.025))/(2*qnorm(0.975)))
n.total <- Fibrinogen$N.total
d.total <- Fibrinogen$N.events

fat(b=b, b.se=b.se)
fat(b=b, b.se=b.se, d.total=d.total, method="D-FIV")

# Note that many tests are also available via metafor
require(metafor)
fat(b=b, b.se=b.se, n.total=n.total, method="M-FIV")
regtest(x=b, sei=b.se, ni=n.total, model="lm", predictor="ni")
```

`Fibrinogen` *Meta-analysis of the association between plasma fibrinogen concentration and the risk of coronary heath disease*

Description

The `Fibrinogen` data set is a meta-analysis of 31 studies in which the association between plasma fibrinogen concentration and the risk of coronary heath disease (CHD) was estimated.

Usage

```
data("Fibrinogen")
```

Format

A data frame with 5 variables:

`N.total` a numeric vector describing the total number of patients for each study

`N.events` a numeric vector describing the number of observed events within each study

`HR` a numeric vector describing the estimated hazard ratio of each study

`HR.025` a numeric vector describing the lower boundary of the 95% confidence interval of `HR`

`HR.975` a numeric vector describing the upper boundary of the 95% confidence interval of `HR`

Source

Fibrinogen Studies Collaboration. Collaborative meta-analysis of prospective studies of plasma fibrinogen and cardiovascular disease. *Eur J Cardiovasc Prev Rehabil*. 2004 Feb;11(1):9-17.

Thompson S, Kaptoge S, White I, Wood A, Perry P, Danesh J, et al. Statistical methods for the time-to-event analysis of individual participant data from multiple epidemiological studies. *Int J Epidemiol*. 2010 Oct;39(5):1345-59.

Examples

```
data(Fibrinogen)
## maybe str(Fibrinogen) ; plot(Fibrinogen) ...
```

`forest` *Forest plot*

Description

Generate a forest plot by specifying the various effect sizes, confidence intervals and summary estimate.

Usage

```
forest(theta, theta.ci.lb, theta.ci.ub, theta.slab, theta.summary,
  theta.summary.ci.lb, theta.summary.ci.ub, theta.summary.pi.lb,
  theta.summary.pi.ub, title, sort = "asc", theme = theme_bw(),
  predint.linetype = 1, xlim, xlab = "", refline = 0,
  label.summary = "Summary Estimate", label.predint = "Prediction Interval",
  ...)
```

Arguments

`theta`	Numeric vector with effect size for each study
`theta.ci.lb`	Numeric vector specifying the lower bound of the confidence interval of the effect sizes
`theta.ci.ub`	Numeric vector specifying the upper bound of the confidence interval of the effect sizes

`theta.slab`	Character vector specifying the study labels
`theta.summary`	Meta-analysis summary estimate of the effect sizes
`theta.summary.ci.lb`	Lower bound of the confidence (or credibility) interval of the summary estimate
`theta.summary.ci.ub`	Upper bound of the confidence (or credibility) interval of the summary estimate
`theta.summary.pi.lb`	Lower bound of the (approximate) prediction interval of the summary estimate.
`theta.summary.pi.ub`	Upper bound of the (approximate) prediction interval of the summary estimate.
`title`	Title of the forest plot
`sort`	By default, studies are sorted by ascending effect size (`sort="asc"`). Set to `"desc"` for sorting in reverse order, or any other value to ignore sorting.
`theme`	Theme to generate the forest plot. By default, the classic dark-on-light ggplot2 theme is used. See theme_bw for more information.
`predint.linetype`	The linetype of the prediction interval
`xlim`	The x limits `(x1, x2)` of the forest plot
`xlab`	Optional character string specifying the X label
`refline`	Optional numeric specifying a reference line
`label.summary`	Optional character string specifying the label for the summary estimate
`label.predint`	Optional character string specifying the label for the (approximate) prediction interval
`...`	Additional arguments, which are currently ignored.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

`Framingham` *Predictive performance of the Framingham Risk Score in male populations*

Description

This data set contains estimates on the performance of the Framingham model for predicting coronary heart disease in male populations (Wilson 1998). Results are based on the original development study and 20 validations identified by Damen *et al*.

Usage

```
data("Framingham")
```

Format

A data frame with 24 observations on the following 19 variables.

`AuthorYear` a vector describing the study authors

`n` a numeric vector with the total number of patients on which performance estimates are based

`n.events` a numeric vector with the total number of observed events

`c.index` a numeric vector with the estimated concordance statistic of each validation

`se.c.index` a numeric vector with the standard error of the concordance statistics

`c.index.95CIl` a numeric vector with the lower bound of the 95% confidence interval of the estimated concordance statistics

`c.index.95CIu` a numeric vector with the upper bound of the 95% confidence interval of the estimated concordance statistics

`Po` a numeric vector with the overall observed event probability of each validation

`Pe` a numeric vector with the overall expected event probability of each validation

`t.val` a numeric vector describing the time period in which predictive performance was assessed for each validation

`mean_age` a numeric vector describing the mean age of the patients

`sd_age` a numeric vector with the spread of the age of the patients

`mean_SBP` a numeric vector with the mean systolic blood pressure in the validation studies (mm Hg)

`sd_SBP` a numeric vector with the spread of systolic blood pressure in the validation studies

`mean_total_cholesterol` a numeric vector with the mean total cholesterol in the validation studies (mg/dL)

`sd_total_cholesterol` a numeric vector with the spread of total cholesterol in the validation studies

`mean_hdl_cholesterol` a numeric vector with the mean high-density lipoprotein cholesterol in the validation studies (mg/dL)

`sd_hdl_cholesterol` a numeric vector with the spread of high-density lipoprotein cholesterol in the validation studies

`pct_smoker` a numeric vector with the percentage smokers in the validation studies

Details

The Framingham Risk Score allows physicians to predict 10-year coronary heart disease (CHD) risk in patients without overt CHD. It was developed in 1998 from a middle-aged white population sample, and has subsequently been validated across different populations. The current dataset contains the original (internal validation) results, as well as 23 external validations which were identified through a systematic review. In this review, studies were eligible for inclusion if they described the validation of the original Framingham model and assessed its performance for fatal or nonfatal CHD in males from a general population setting.

Source

Damen JAAG, Hooft L, Schuit E, Debray TPA, Collins GS, Tzoulaki I, et al. Prediction models for cardiovascular disease risk in the general population: systematic review. *BMJ*. 2016;i2416.

Wilson PW, D'Agostino RB, Levy D, Belanger AM, Silbershatz H, Kannel WB. Prediction of coronary heart disease using risk factor categories. *Circulation*. 1998; **97**(18):1837–47.

Examples

```
data(Framingham)
```

`inv.logit` *Apply the inverse logit tranformation*

Description

Transforms a linear predictor into a probability.

Usage

```
inv.logit(x)
```

Arguments

`x`	A vector of numerics (between -Inf and Inf)

Value

A vector of numerics between 0 and 1.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

See Also

`logit`

Kertai *Kertai data*

Description

Data frame with diagnostic accuracy data from exercise electrocardiography.

Usage

```
data("Kertai")
```

Format

One data frame with 4 variables.

TP integer. number of true positives

FN integer. number of false negatives

FP integer. number of false positives

TN integer. number of true negatives

Details

The `Kertai` data set is a meta-analysis of prognostic test studies and comprises 7 studies where the diagnostic test accuracy of exercise electrocardiography for predicting cardiac events in patients undergoing major vascular surgery was measured.

Source

Kertai MD, Boersma E, Bax JJ, Heijenbrok-Kal MH, Hunink MGM, L'talien GJ, Roelandt JRTC, van Urk H, Poldermans D. A meta-analysis comparing the prognostic accuracy of six diagnostic tests for predicting perioperative cardiac risk in patients undergoing major vascular surgery. *Heart* 2003; **89**: 1327–1334.

Jackson D, Riley RD, & White IW. Multivariate meta-analysis: Potential and promise. *Statistics in Medicine* 2010; **30**: 2481–2498.

logit *Apply logit tranformation*

Description

Transforms values between 0 and 1 to values between -Inf and Inf.

Usage

```
logit(x)
```

Arguments

x	A vector of numerics (between 0 and 1)

Value

A vector of numerics (between -Inf and Inf).

Author(s)

Thomas Debray <thomas.debray@gmail.com>

See Also

inv.logit

---

logLik.riley *Print the log-likelihood*

Description

This function provides the (restricted) log-likelihood of a fitted model.

Usage

```
## S3 method for class 'riley'
logLik(object, ...)
```

Arguments

object	A riley object, representing a fitted alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown.
...	Additional arguments to be passed on to other functions, currently ignored.

Value

Returns an object of class logLik. This is the (restricted) log-likelihood of the model represented by object evaluated at the estimated coefficients. It contains at least one attribute, "df" (degrees of freedom), giving the number of (estimated) parameters in the model.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

Riley RD, Thompson JR, Abrams KR. An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown. *Biostatistics* 2008; **9**: 172–186.

Examples

```
data(Daniels)
fit <- riley(Daniels,control=list(maxit=10000))
logLik(fit)
```

`metapred` *Generalized Stepwise Regression for prediction models*

Description

Generalized stepwise regression for obtaining a prediction model with adequate performance across data sets. Requires data from individuals in multiple studies.

Usage

```
metapred(data, strata, formula = NULL, estFUN = "glm", stepwise = TRUE,
  center.out = FALSE, center.cov = FALSE, recal.int = FALSE,
  cvFUN = NULL, cv.k = NULL, metaFUN = NULL, meta.method = "REML",
  predFUN = NULL, perfFUN = NULL, genFUN = NULL, selFUN = "which.min",
  ...)
```

Arguments

`data`	data.frame containing the datasets.
`strata`	Name of the strata (e.g. studies or clusters) variable, as character. Used for two-stage MA only.
`formula`	Formula of the full model to be evaluated, and possibly reduced. If not supplied, it is assumed the first column in the data set is the outcome, and all remaining columns (except `strata`) are predictors. See formula for details.
`estFUN`	Function for estimating the model in the first stage. Currently "lm" and "glm" are supported.
`stepwise`	Logical. Should stepwise selection be performed?
`center.out`	Logical. Should the outcome be centered within studies?
`center.cov`	Logical. Should covariates be centered within studies?
`recal.int`	Logical. Should the intercept be recalibrated?
`cvFUN`	Cross-validation method, on the study (i.e. cluster or stratum) level. " l1o" for leave-one-out cross-validation (default). "bootstrap" for bootstrap. Or "fixed", for one or more data sets which are only used for validation. A user written function may be supplied as well.
`cv.k`	Parameter for cvFUN. For `cvFUN="bootstrap"`, this is the number of bootstraps. For `cvFUN="fixed"`, this is a vector of the indices of the (sorted) data sets.

metaFUN	Function for computing the meta-analytic coefficient estimates in two-stage MA. Default: rma from the metafor package is used. Default settings are univariate random effects, estimated with "REML". Method can be passed trough the meta.method argument.
meta.method	Name of method for meta-analysis. Default is "REML". For more options see rma.
predFUN	Function for predicting new values. Defaults to the appropriate link functions for two-stage MA where glm() or lm() is used in the first stage. For one-stage models predict() is used.
perfFUN	Function for computing the performance of the prediction models. Default: mean squared error (perfFUN="mse"). Other options are "vare".
genFUN	Function computing generalizability measure using the performance measures. Default: (absolute) mean (genFUN="absmean"). Choose squareddiff for a penalty equal to the mean squared differences between coefficients. Alternatively, choose pooledvar for a weighted average of variance terms.
selFUN	Function for selecting the best method. Default: lowest value for genFUN. Should be set to "which.max" to indicate a good model.
...	To pass arguments to estFUN (e.g. family = "binomial"), or other methods.

Value

metapred A list of class metapred, containing the final coefficients in coefficients, and the stepwise tree of estimates of the coefficients (coef), performance measures (perf), generalizability measures (gen) in stepwise, and more.

Author(s)

Valentijn de Jong

References

Debray TPA, Moons KGM, Ahmed I, Koffijberg H, Riley RD. A framework for developing, implementing, and evaluating clinical prediction models in an individual participant data meta-analysis. *Stat Med*. 2013;32(18):3158-80.

Examples

```
data(DVTipd)
DVTipd$cluster <- 1:4 # Add a fictional clustering to the data set.
metamisc:::metapred(DVTipd, strata = "cluster", f = dvt ~ sex + vein + malign, family = binomial)

## Not run:
# Some additional examples:
metamisc:::metapred(DVTipd, strata = "cluster", f = dvt ~ sex + vein + malign
, family = binomial, stepwise = FALSE)
metamisc:::metapred(DVTipd, strata = "cluster", f = dvt ~ sex + altdiagn + histdvt
, family = binomial, recal.int = TRUE)
metamisc:::metapred(DVTipd, strata = "cluster", f = dvt ~ sex + altdiagn + histdvt
, family = binomial, meta.method = "DL")
```

```
## End(Not run)
# By default, metapred assumes the first column is the outcome.
DVTipd.reordered <- DVTipd[c("dvt", "ddimdich", "histdvt", "cluster")]
mp <- metamisc:::metapred(DVTipd.reordered, strata = "cluster", family = binomial)
fitted <- predict(mp, newdata = DVTipd.reordered)
```

`oecalc` *Calculate the total O:E ratio*

Description

This function calculates (transformed versions of) the ratio of total number of observed versus expected events with the corresponding sampling variance.

Usage

```
oecalc(OE, OE.se, OE.cilb, OE.ciub, OE.cilv, citl, citl.se, N, O, E, Po, Po.se,
  Pe, data, slab, add = 1/2, g = NULL, level = 0.95, ...)
```

Arguments

`OE`	vector with the estimated ratio of total observed versus total expected events
`OE.se`	vector with the standard errors of the estimated O:E ratios
`OE.cilb`	vector to specify the lower limits of the confidence interval for `OE`.
`OE.ciub`	vector to specify the upper limits of the confidence interval for `OE`.
`OE.cilv`	vector to specify the levels of aformentioned confidence interval limits. (default: 0.95, which corresponds to the 95% confidence interval).
`citl`	vector with the estimated calibration-in-the-large statistics
`citl.se`	vector with the standard error of the calibration-in-the-large statistics
`N`	vector to specify the sample/group sizes.
`O`	vector to specify the total number of observed events.
`E`	vector to specify the total number of expected events
`Po`	vector to specify the (cumulative) observed event probabilities.
`Po.se`	vector with the standard errors of `Po`.
`Pe`	vector to specify the (cumulative) expected event probabilites (if specified, during time `t.val`)
`data`	optional data frame containing the variables given to the arguments above.
`slab`	optional vector with labels for the studies.
`add`	a non-negative number indicating the amount to add to zero counts. See 'Details'
`g`	a quoted string that is the function to transform estimates of the total O:E ratio; see the details below.
`level`	level for confidence interval, default `0.95`.
`...`	Additional arguments.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

Examples

```
######### Validation of prediction models with a binary outcome #########
data(EuroSCORE)

# Calculate the total O:E ratio and its standard error
oecalc(O=n.events, E=e.events, N=n, data=EuroSCORE, slab=Study)

# Calculate the log of the total O:E ratio and its standard error
oecalc(O=n.events, E=e.events, N=n, data=EuroSCORE, slab=Study, g="log(OE)")
```

---

plot.fat *Display results from the funnel plot asymmetry test*

Description

Generates a funnel plot for a fitted `fat` object.

Usage

```
## S3 method for class 'fat'
plot(x, ref, confint = TRUE, confint.level = 0.1,
  confint.col = "skyblue", confint.density = NULL, xlab = "Effect size",
  ...)
```

Arguments

`x`	An object of class `fat`
`ref`	A numeric value indicating the fixed or random effects summary estimate. If no value is provided then it will be retrieved from a fixed effects meta-analysis (if possible).
`confint`	A logical indicator. If `TRUE`, a confidence interval will be displayed for the estimated regression model (based on a Student-T distribution)
`confint.level`	Significance level for constructing the confidence interval.
`confint.col`	The color for filling the confidence interval. Choose `NA` to leave polygons unfilled. If `confint.density` is specified with a positive value this gives the color of the shading lines.
`confint.density`	The density of shading lines, in lines per inch. The default value of `NULL` means that no shading lines are drawn. A zero value of density means no shading nor filling whereas negative values and `NA` suppress shading (and so allow color filling).
`xlab`	A title for the x axis
`...`	Additional arguments for `plot`.

Examples

```
data(Fibrinogen)
b <- log(Fibrinogen$HR)
b.se <- ((log(Fibrinogen$HR.975) - log(Fibrinogen$HR.025))/(2*qnorm(0.975)))
n.total <- Fibrinogen$N.total

# A very simple funnel plot
plot(fat(b=b, b.se=b.se))

# Add custom tickmarks for the X-axis
plot(fat(b=b, b.se=b.se, n.total=n.total, method="M-FIV"), xlab="Hazard ratio", xaxt="n")
axis(1, at=c(log(0.5), log(1), log(1.5), log(2), log(3)), labels=c(0.5, 1, 1.5, 2,3))
```

---

plot.riley *Plot the summary of the bivariate model from Riley et al. (2008).*

Description

Generates a forest plot for each outcome of the bivariate meta-analysis.

Usage

```
## S3 method for class 'riley'
plot(x, title, sort = "asc", xlim, refline, ...)
```

Arguments

x	An object of class riley
title	Title of the forest plot
sort	By default, studies are ordered by ascending effect size (sort="asc"). For study ordering by descending effect size, choose sort="desc". For any other value, study ordering is ignored.
xlim	The x limits (x1, x2) of the forest plot
refline	Optional numeric specifying a reference line
...	Additional parameters for generating forest plots

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

Riley RD, Thompson JR, Abrams KR. An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown. *Biostatistics* 2008; **9**: 172–186.

Examples

```
data(Scheidler)

#Perform the analysis
fit <- riley(Scheidler[which(Scheidler$modality==1),])
plot(fit)

require(ggplot2)
plot(fit, sort="none", theme=theme_gray())
```

`plot.uvmeta` *Forest Plots*

Description

Function to create forest plots for objects of class `"uvmeta"`.

Usage

```
## S3 method for class 'uvmeta'
plot(x, sort = "asc", ...)
```

Arguments

`x`	An object of class `"uvmeta"`
`sort`	By default, studies are ordered by ascending effect size (`sort="asc"`). For study ordering by descending effect size, choose `sort="desc"`. For any other value, study ordering is ignored.
`...`	Additional arguments which are passed to forest.

Details

The forest plot shows the performance estimates of each validation with corresponding confidence intervals. A polygon is added to the bottom of the forest plot, showing the summary estimate based on the model. A 95% prediction interval is added by default for random-effects models, the dotted line indicates its (approximate) bounds

Note

Full lines indicate confidence intervals or credibility intervals (in case of a Bayesian meta-analysis). Dashed lines indicate prediction intervals. The width of all intervals is defined by the significance level chosen during meta-analysis.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

- Lewis S, Clarke M. Forest plots: trying to see the wood and the trees. *BMJ*. 2001; 322(7300):1479–80.
- Riley RD, Higgins JPT, Deeks JJ. Interpretation of random effects meta-analyses. *BMJ*. 2011 342:d549–d549.

Examples

```
data(Roberts)

# Frequentist random-effects meta-analysis
fit <- with(Roberts, uvmeta(r=SDM, r.se=SE, labels=rownames(Roberts)))
plot(fit)
```

`plot.valmeta` *Forest Plots*

Description

Function to create forest plots for objects of class `"valmeta"`.

Usage

```
## S3 method for class 'valmeta'
plot(x, sort = "asc", ...)
```

Arguments

`x`	An object of class `"valmeta"`
`sort`	By default, studies are ordered by ascending effect size (`sort="asc"`). For study ordering by descending effect size, choose `sort="desc"`. For any other value, study ordering is ignored.
`...`	Additional arguments which are passed to forest.

Details

The forest plot shows the performance estimates of each validation with corresponding confidence intervals. A polygon is added to the bottom of the forest plot, showing the summary estimate based on the model. A 95% prediction interval is added by default for random-effects models, the dotted line indicates its (approximate) bounds.

Value

An object of class `ggplot`

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

- Debray TPA, Damen JAAG, Snell KIE, Ensor J, Hooft L, Reitsma JB, et al. A guide to systematic review and meta-analysis of prediction model performance. *BMJ*. 2017;356:i6460.
- Lewis S, Clarke M. Forest plots: trying to see the wood and the trees. *BMJ*. 2001; 322(7300):1479–80.
- Riley RD, Higgins JPT, Deeks JJ. Interpretation of random effects meta-analyses. *BMJ*. 2011 342:d549–d549.

Examples

```
data(EuroSCORE)
fit <- with(EuroSCORE, valmeta(cstat=c.index, cstat.se=se.c.index,
            cstat.95CI=cbind(c.index.95CIl,c.index.95CIu), N=n, O=n.events))
plot(fit)

library(ggplot2)
plot(fit, theme=theme_grey())
```

`predict.riley` *Prediction Interval*

Description

Calculates a prediction interval for the summary parameters of Riley's alternative model for bivariate random-effects meta-analysis. This interval predicts in what range future observations will fall given what has already been observed.

Usage

```
## S3 method for class 'riley'
predict(object, ...)
```

Arguments

`object`	A `riley` object.
`...`	Additional arguments (currently ignored)

Details

Prediction intervals are based on Student's t-distribution with (numstudies - 5) degrees of freedom. The width of the interval is specified by the significance level chosen during meta-analysis.

Value

Data frame containing prediction intervals with the summary estimates `beta1` and `beta2` (for effect size data), or with the mean sensitivity and false positive rate (for diagnostic test accuracy data).

Author(s)

Thomas Debray <thomas.debray@gmail.com>

`recalibrate` *Recalibrate a Prediction Model*

Description

`recalibrate` is used to recalibrate a prediction model of classes `metapred, glm` or `lm`.

Usage

```
recalibrate(object, newdata, f = ~1, estFUN = NULL, ...)
```

Arguments

`object`	A model fit object to be recalibrated, of class `metapred, glm` or `lm`, and more.
`newdata`	data.frame containing new data set for updating.
`f`	formula. Which coefficients of the model should be updated? Default: intercept only. Left-hand side may be left out. See formula for details.
`estFUN`	Function for model estimation. If left `NULL`, the function is automatically retrieved for `metapred` objects. For other objects, the function with name corresponding to the first class of the object is taken. E.g. `glm()` for `glm` objects.
`...`	Optional arguments to pass to `estFUN`.

Details

Currently only the coefficients are updated and the variances and other aspects are left untouched. For updating the entire model and all its statistics, see update.

Value

Recalibrated model fit object, of the same class as `object`. Generally, updated coefficients can be retrieved with `coef()`.

Examples

```
data(DVTipd)
DVTipd$cluster <- 1:4 # Add a fictional clustering to the data set.
# Suppose we estimated the model in three studies:
DVTipd123 <- DVTipd[DVTipd$cluster <= 3, ]
mp <- metamisc:::metapred(DVTipd123, strata = "cluster", f = dvt ~ vein + malign,
family = binomial)
# and now want to recalibrate it for the fourth:
DVTipd4 <- DVTipd[DVTipd$cluster == 4, ]
metamisc:::recalibrate(mp, newdata = DVTipd4)
```

`riley` *Fit the alternative model for bivariate random-effects meta-analysis*

Description

This function fits the alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown. This bivariate model was proposed by Riley et al. (2008) and is similar to the general bivariate random-effects model (van Houwelingen et al. 2002), but includes an overall correlation parameter rather than separating the (usually unknown) within- and between-study correlation. As a consequence, the alternative model is not fully hierarchical, and estimates of additional variation beyond sampling error (`psi`) are not directly equivalent to the between-study variation (`tau`) from the general model. This model is particularly useful when there is large within-study variability, few primary studies are available or the general model estimates the between-study correlation as 1 or -1.

Usage

```
riley(X, slab, optimization = "Nelder-Mead", control = list(), pars, ...)
```

Arguments

`X`	data frame containing integer variables `Y1`, `vars1`, `Y2` and `vars2`, where the columns `Y1` and `Y2` represent the effect sizes of outcome 1 and, respectively, outcome 2. The columns `vars1` and `vars2` represent the error variances of `Y1` and, respectively, `Y2`. Alternatively, when considering a meta-analysis of diagnostic test accuracy data, the columns `TP`, `FN`, `FP` and `TN` may be specified. Corresponding values then represent the number of true positives, the number of false negatives, the number of false positives and, respectively, the number of true negatives.
`slab`	Optional vector specifying the label for each study
`optimization`	The optimization method that should be used for minimizing the negative (restricted) log-likelihood function. The default method is an implementation of that of Nelder and Mead (1965), that uses only function values and is robust but relatively slow. Other methods are described in optim.
`control`	A list of control parameters to pass to optim.

pars	List with additional arguments. The width of confidence, credibility and prediction intervals is defined by level (defaults to 0.95).
...	Arguments to be passed on to other functions. See "Details" for more information.

Details

Parameters are estimated by iteratively maximizing the restriced log-likelihood using the Newton-Raphson procedure. The results from a univariate random-effects meta-analysis with a method-of-moments estimator are used as starting values for beta1, beta2, psi1 and psi2 in the optim command. Standard errors for all parameters are obtained from the inverse Hessian matrix.

Meta-analysis of effect sizes: The following parameters are estimated by iteratively maximizing the restriced log-likelihood using the Newton-Raphson procedure: pooled effect size for outcome 1 (beta1), pooled effect size for outcome 2 (beta2), additional variation of beta1 beyond sampling error (psi1), additional variation of beta2 beyond sampling error (psi2) and the correlation rho between psi1 and psi2.

Meta-analysis of diagnostic test accuracy: Although the model can also be used for diagnostic test accuracy data when substantial within-study correlations are expected, assuming zero within-study correlations (i.e. applying Reitsma's approach) is usually justified (Reitsma et al. 2005, Daniels and Hughes 1997, Korn et al. 2005, Thompson et al. 2005, Van Houwelingen et al. 2002).

A logit transformation is applied to the sensitivities ans false positive rates of each study, in order to meet the normality assumptions. When zero cell counts occur, continuity corrections may be required. The correction value can be specified using correction (defaults to 0.5). Further, when the argument correction.control is set to "all" (the default) the continuity correction is added to the whole data if only one cell in one study is zero. If correction.control="single" the correction is only applied to rows of the data which have a zero.

The following parameters are estimated: logit of sensitivity (beta1), logit of false positive rate (beta2), additional variation of beta1 beyond sampling error (psi1), additional variation of beta2 beyond sampling error (psi2) and the correlation (rho) between psi1 and psi2.

Value

An object of the class riley for which many standard methods are available. A warning message is casted when the Hessian matrix contains negative eigenvalues, which implies that the identified solution is a saddle point and thus not optimal.

Note

The overall correlation parameter rho is transformed during estimation to ensure that corresponding values remain between -1 and 1. The transformed correlation rhoT is given as logit((rho+1)/2). During optimization, the starting value for rhoT is set to 0. The standard error of rho is derived from rhoT using the Delta method. Similarly, the Delta methods is used to derive the standard error of the sensitivity and false positive rate from beta1 and, respectively, beta2.

Algorithms for dealing with missing data are currently not implemented, but Bayesian approaches will become available in later versions.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

- Korn EL, Albert PS, McShane LM. Assessing surrogates as trial endpoints using mixed models. *Statistics in Medicine* 2005; **24**: 163–182.
- Nelder JA, Mead R. A simplex algorithm for function minimization. *Computer Journal* (1965); **7**: 308–313.
- Reitsma J, Glas A, Rutjes A, Scholten R, Bossuyt P, Zwinderman A. Bivariate analysis of sensitivity and specificity produces informative summary measures in diagnostic reviews. *Journal of Clinical Epidemiology* 2005; **58**: 982–990.
- Riley RD, Thompson JR, Abrams KR. An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown. *Biostatistics* 2008; **9**: 172–186.
- Thompson JR, Minelli C, Abrams KR, Tobin MD, Riley RD. Meta-analysis of genetic studies using mendelian randomization–a multivariate approach. *Statistics in Medicine* 2005; **24**: 2241–2254.
- van Houwelingen HC, Arends LR, Stijnen T. Advanced methods in meta-analysis: multivariate approach and meta-regression. *Statistics in Medicine* 2002; **21**: 589–624.

Examples

```
data(Scheidler)
data(Daniels)
data(Kertai)

# Meta-analysis of potential surrogate markers data
# The results obtained by Riley (2008) were as follows:
# beta1 = -0.042 (SE = 0.063),
# beta2 = 14.072 (SE = 4.871)
# rho   = -0.759
## Not run:
fit1 <- riley(Daniels) #maxit reached, try again with more iterations

## End(Not run)
fit1 <- riley(Daniels, control=list(maxit=10000))
summary(fit1)

# Meta-analysis of prognostic test studies
fit2 <- riley(Kertai)
fit2

# Meta-analysis of computed tomography data
ds <- Scheidler[which(Scheidler$modality==1),]
fit3 <- riley(ds)
fit3
```

Roberts *Roberts data*

Description

Data frame with summary data from 14 comparative studies.

Usage

```
data("Roberts")
```

Format

One data frame with 2 variables.

SDM Effect sizes (standardized differences in means)

SE Standard error of the effect sizes

Details

The `Roberts` data set is a meta-analysis of 14 studies comparing 'set shifting' ability (the ability to move back and forth between different tasks) in people with eating disorders and healthy controls.

Source

Roberts ME, Tchanturia K, Stahl D, Southgate L, Treasure J. A systematic review and meta-analysis of set-shifting ability in eating disorders. *Psychological Medicine* 2007, **37**: 1075–1084.

Higgins JPT, Thompson SG, Spiegelhalter DJ. A re-evaluation of random-effects meta-analysis. *Journal of the Royal Statistical Society. Series A (Statistics in Society)* 2009, **172**: 137–159.

Scheidler *Diagnostic accuracy data*

Description

Data frame with diagnostic accuracy data from three imaging techniques for the diagnosis of lymph node metastasis in women with cervical cancer.

Usage

```
data("Scheidler")
```

Format

One data frame with 6 variables.

author string . author of article

modality integer . type of test (1=CT, 2=LAG, 3=MRI)

TP integer. number of true positives

FN integer. number of false negatives

FP integer. number of false positives

TN integer. number of true negatives

Details

The `Scheidler` data comprises the results from a meta-analysis where three imaging techniques for the diagnosis of lymph node metastasis in women with cervical cancer are compared. Forty-four studies in total were included: 17 studies evaluated lymphangiography, another 17 studies examined computed tomography and the remaining 10 studies focused on magnetic resonance imaging. Diagnosis of metastatic disease by lymphangiography (LAG) is based on the presence of nodal-filling defects, whereas computed tomography (CT) and magnetic resonance imaging (MRI) rely on nodal enlargement.

Source

Scheidler J, Hricak H, Yu KK, Subak L, Segal MR. Radiological evaluation of lymph node metastases in patients with cervical cancer. A meta-analysis. *Journal of the American Medical Association* 1997; **278**: 1096–1101.

Reitsma J, Glas A, Rutjes A, Scholten R, Bossuyt P, Zwinderman A. Bivariate analysis of sensitivity and specificity produces informative summary measures in diagnostic reviews. *Journal of Clinical Epidemiology* 2005; **58**: 982–990.

`stackedglm` *Stacked Regression*

Description

This function combines one or more existing prediction models into a so/called meta-model.

Usage

```
stackedglm(models, family = binomial, data)
```

Arguments

models	a list containing the historical prediction models, which can be defined in several ways. For instance, historical regression models can be specified using a named vector containing the regression coefficients of the individual predictors (no need to include the intercept term). List items may also represent an object for which the function predict() exists.
family	a description of the error distribution and link function to be used in the meta-model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See family for details of family functions.)
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which stackedglm is called.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

---

summary.riley *Parameter summaries Provides the summary estimates of the alternative model for bivariate random-effects meta-analysis by Riley et al. (2008) with their corresponding standard errors (derived from the inverse Hessian). For confidence intervals, asymptotic normality is assumed.*

Description

Parameter summaries Provides the summary estimates of the alternative model for bivariate random-effects meta-analysis by Riley et al. (2008) with their corresponding standard errors (derived from the inverse Hessian). For confidence intervals, asymptotic normality is assumed.

Usage

```
## S3 method for class 'riley'
summary(object, ...)
```

Arguments

object	A riley object
...	Arguments to be passed on to other functions (currently ignored)

Details

For meta-analysis of diagnostic test accuracy data, beta1 equals the logit sensitivity (Sens) and beta2 equals the logit false positive rate (FPR).

Value

array with confidence intervals for the estimated model parameters. For diagnostic test accuracy data, the resulting summary sensitivity and false positive rate are included.

Note

For the overall correlation (`rho`) confidence intervals are derived using the transformation `logit((rho+1)/2)`. Similarly, the logit transformation is used to derive confidence intervals for the summary sensitivity and false positive rate.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

Riley RD, Thompson JR, Abrams KR. An alternative model for bivariate random-effects meta-analysis when the within-study correlations are unknown. *Biostatistics* 2008; **9**: 172–186.

`summary.uvmeta` *Summarizing Univariate Meta-Analysis Models*

Description

This function provides summary estimates of a fitted univariate meta-analysis model.

Usage

```
## S3 method for class 'uvmeta'
summary(object, ...)
```

Arguments

`object`	An object of class `"uvmeta"`
`...`	Optional arguments to be passed on to other functions

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

- Borenstein M, Hedges LV, Higgins JPT, Rothstein HR. A basic introduction to fixed-effect and random-effects models for meta-analysis. *Research Synthesis Methods* 2010; **1**: 97–111.
- DerSimonian R, Laird N. Meta-analysis in clinical trials. *Controlled Clinical Trials* 1986; **7**: 177–188.
- Riley RD, Higgins JPT, Deeks JJ. Interpretation of random effects meta-analyses. *British Medical Journal* 2011; **342**: d549.

See Also

uvmeta

---

uvmeta *Univariate meta-analysis*

Description

This function summarizes multiple estimates for a single parameter by assuming a fixed (i.e. common) effect or random effects across studies. The summary estimate is obtained by calculating a weighted mean that accounts for sample size and (in case random effects are assumed) for between-study heterogeneity.

Usage

```
uvmeta(r, r.se, method = "REML", test = "knha", labels, na.action,
  n.chains = 4, pars, ret.fit = FALSE, verbose = FALSE, ...)
```

Arguments

r	Vector of numerics containing the effect size of each study
r.se	Vector of numerics containing the standard error of the effect sizes
method	Character string specifying whether a fixed-effect or a random-effects model should be fitted. A fixed-effect model is fitted when using method="FE". Random-effects models are fitted by setting method equal to one of the following: "REML" (Default), "DL", "HE", "SJ", "ML", "EB", "HS", "GENQ" or "BAYES". See 'Details'.
test	Optional character string when method!="BAYES" to specify how test statistics and confidence intervals for the fixed effects should be computed. By default (test="knha"), the method by Knapp and Hartung (2003) is used for adjusting test statistics and confidence intervals. Type '?rma' for more details.
labels	Vector of characters containing the labels for the studies
na.action	A function which indicates what should happen when the data contain NAs. Defaults to "na.fail", other options are "na.omit", "na.exclude" or "na.pass".
n.chains	Optional numeric specifying the number of chains to use in the Gibbs sampler (method="BAYES"). More chains will improve the sensitivity of the convergence diagnostic, but will cause the simulation to run more slowly. The default number of chains is 4.
pars	Optional list with additional arguments. The width of confidence, credibility and prediction intervals is defined by level (defaults to 0.95). The following parameters configure the MCMC sampling procedure: hp.mu.mean (Hyperparameter: mean of the prior distribution of the fixed/random effects model, defaults to zero), hp.mu.var (Hyperparameter: variance of the prior distribution of the fixed/random effects model, defaults to 1000), hp.tau.min (Hyperparameter: mininum value for the between-study standard deviation, defaults to 0),

	hp.tau.max (Hyperparameter: maximum value for the between-study standard deviation, defaults to 100).
ret.fit	logical indicating whether the full results from the fitted model should also be returned.
verbose	If TRUE then messages generated during the fitting process will be displayed.
...	Additional arguments that are passed to **rma** or **runjags** (if method="BAYES").

Details

Unless specified otherwise, all meta-analysis models assume random effects and are fitted using restricted maximum likelihood estimation with the **metafor** package (Viechtbauer 2010). Further, confidence intervals for the average performance are based on the Hartung-Knapp-Sidik-Jonkman method, to better account for the uncertainty in the estimated between-study heterogeneity (Debray 2016). A Bayesian meta-analysis can be performed by specifying method="BAYES". In that case, the R packages **runjags** and **rjags** must be installed.]

For random-effects models, a prediction interval for the pooled effect size is displayed. This interval predicts in what range future effect sizes will fall given what has already been observed (Higgins 2009, Riley 2011).

Bayesian meta-analysis models: For Bayesian meta-analysis models that involve the Gibbs sampler (method="BAYES"), the R packages runjags and rjags must be installed. The Bayesian approach uses an uninformative Normal prior for the mean and a uniform prior for the between-study variance of the pooled effect size (Higgins 2009). By default, the Normal prior has a mean of 0 and a variance of 1000. These hyperparameters can, however, be altered through the variables hp.mu.mean and hp.mu.var in the argument pars. The prior distribution of the between-study standard deviation is given by a uniform distribution, by default bounded between 0 and 100.

Value

An object of the class uvmeta for which many standard methods are available.

"data" array with (transformed) data used for meta-analysis, and method(s) used for restoring missing information.

"method" character string specifying the meta-analysis method.

"est" estimated performance statistic of the model. For Bayesian meta-analysis, the posterior median is returned.

"se" standard error (or posterior standard deviation) of the summary estimate.

"tau2" estimated amount of (residual) heterogeneity. Always 0 when method="FE". For Bayesian meta-analysis, the posterior median is returned.

"se.tau2" estimated standard error (or posterior standard deviation) of the between-study variation.

"ci.lb" lower bound of the confidence (or credibility) interval of the summary estimate

"ci.ub" upper bound of the confidence (or credibility) interval of the summary estimate

"pi.lb" lower bound of the (approximate) prediction interval of the summary estimate

"pi.ub" upper bound of the (approximate) prediction interval of the summary estimate

"fit" the full results from the fitted model

"slab" vector specifying the label of each study.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

- Biggerstaff BJ, Tweedie RL. Incorporating variability in estimates of heterogeneity in the random effects model in meta-analysis. *Statistics in Medicine* 1997; **16**: 753–768.
- Borenstein M, Hedges LV, Higgins JPT, Rothstein HR. A basic introduction to fixed-effect and random-effects models for meta-analysis. *Research Synthesis Methods* 2010; **1**: 97–111.
- DerSimonian R, Laird N. Meta-analysis in clinical trials. *Controlled Clinical Trials* 1986; **7**: 177–188.
- Graham PL, Moran JL. Robust meta-analytic conclusions mandate the provision of prediction intervals in meta-analysis summaries. *Journal of Clinical Epidemiology* 2012; **65**: 503–510.
- Higgins JPT, Thompson SG. Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine* 2002; **21**: 1539–1558.
- Higgins JPT, Thompson SG, Spiegelhalter DJ. A re-evaluation of random-effects meta-analysis. *Journal of the Royal Statistical Society. Series A (Statistics in Society)* 2009, **172**: 137–159.
- Riley RD, Higgins JPT, Deeks JJ. Interpretation of random effects meta-analyses. *British Medical Journal* 2011; **342**: d549.
- Viechtbauer W. Conducting Meta-Analyses in R with the metafor Package. *Journal of Statistical Software*. 2010; 36(3). Available from: http://www.jstatsoft.org/v36/i03/

Examples

```
data(Roberts)

# Frequentist random-effects meta-analysis
fit1 <- with(Roberts, uvmeta(r=SDM, r.se=SE, labels=rownames(Roberts)))
summary(fit1)
plot(fit1) #show a forest plot
fit1

## Not run:
# Bayesian random effects meta-analysis
fit2 <- with(Roberts, uvmeta(r=SDM, r.se=SE, labels=rownames(Roberts), method="BAYES"))
plot(fit2)

## End(Not run)
```

`uvmeta-class` *Class "uvmeta". Result of a univariate meta-analysis.*

Description

This class encapsulates results of a univariate meta-analysis.

Objects from the Class

Objects can be created by calls of the form `uvmeta`.

Slots

`call`: (language) The call to `uvmeta`.

`data`: (data frame) The data used for the meta-analysis.

`results`: (data frame) Contains the pooled effect size (`mu`), the between-study variability (`tausq`), Cochran's Q statistic (`Q`) and Higgins' and Thompson's I square statistic (`Isq`). For each estimate, error variances are provided with predefined confidence (`method="MOM"`) or credibility (`method="bayes"`) intervals.

`model`: (character) The meta-analysis model used.

`method`: (character) The estimator used.

`na.action`: (character) Information from the action which was applied to object if NAs were handled specially, or NULL.

`df`: (numeric) Degrees of freedom.

`numstudies`: (numeric) The amount of studies used in the meta-analysis.

`pred.int`: (data frame) A prediction interval, predicting in what range future effect sizes will fall given what has already been observed (based on a Student's t-distribution, cfr. Riley 2011)

`formula`: (character) If a formula was specified, a character vector giving the formula and parameter specifications.

Methods

print `signature(object = "uvmeta")`: Print object summary.

forest `signature(object = "uvmeta")`: Plot a forest plot with the summary estimate.

summary `signature(object = "uvmeta")`: Generate object summary.

Examples

```
data(Collins)

#Extract effect size and error variance
r <- Collins$logOR
vars <- Collins$SE**2

#Frequentist random-effects meta-analysis
fit1 <- uvmeta(r,vars)

#Extract results
fit1$results
```

valmeta *Meta-analysis of prediction model performance*

Description

This function provides summary estimates for the concordance statistic, the total observed-expected ratio or the calibration slope. Where appropriate, data transformations are applied and missing information is derived from available quantities. Unless specified otherwise, all meta-analysis models assume random effects and are fitted using restricted maximum likelihood estimation with the **metafor** package (Viechtbauer 2010). Further, confidence intervals for the average performance are based on the Hartung-Knapp-Sidik-Jonkman method. When conducting a Bayesian meta-analysis, the R packages **runjags** and **rjags** must be installed.

Usage

```
valmeta(measure = "cstat", cstat, cstat.se, cstat.95CI, sd.LP, OE, OE.se,
  OE.95CI, citl, citl.se, N, O, E, Po, Po.se, Pe, method = "REML",
  test = "knha", ret.fit = FALSE, verbose = FALSE, slab, n.chains = 4,
  pars, ...)
```

Arguments

measure	A character string indicating which summary performance measure should be calculated. Options are "cstat" (meta-analysis of the concordance statistic) and "OE" (meta-analysis of the total observed-expected ratio). See 'Details' for more information.
cstat	Optional vector with the estimated c-statistic for each valiation
cstat.se	Optional vector with the standard error of the estimated c-statistics
cstat.95CI	Optional 2-dimensional array with the lower (first column) and upper (second column) boundary of the 95% confidence interval of the estimated c-statistics
sd.LP	Optional vector with the standard deviation of the linear predictor (prognostic index)
OE	Optional vector with the estimated ratio of total observed versus total expected events
OE.se	Optional vector with the standard errors of the estimated O:E ratios
OE.95CI	Optional 2-dimensional array with the lower (first column) and upper (second column) boundary of the 95% confidence interval of the total O:E ratios
citl	Optional vector with the estimated calibration-in-the-large for each valiation
citl.se	Optional vector with the standard error of the estimated calibration-in-the-large statistics
N	Optional vector with the total number of participants for each valiation
O	Optional vector with the total number of observed events for each valiation (if specified, during time t.val)

E	Optional vector with the total number of expected events for each valiation (if specified, during time t.val)
Po	Optional vector with the (cumulative) observed event probability for each valiation (if specified, during time t.val)
Po.se	Optional vector with the standard errors of Po.
Pe	Optional vector with the (cumulative) expected event probability for each validation (if specified, during time t.val)
method	Character string specifying whether a fixed- or a random-effects model should be fitted. A fixed-effects model is fitted when using method="FE". Random-effects models are fitted by setting method equal to one of the following: "REML" (Default), "DL", "HE", "SJ", "ML", "EB", "HS", "GENQ" or "BAYES". See 'Details'.
test	Optional character string specifying how test statistics and confidence intervals for the fixed effects should be computed. By default (test="knha"), the method by Knapp and Hartung (2003) is used for adjusting test statistics and confidence intervals. Type '?rma' for more details.
ret.fit	logical indicating whether the full results from the fitted model should also be returned.
verbose	If TRUE then messages generated during the fitting process will be displayed.
slab	Optional vector specifying the label for each study
n.chains	Optional numeric specifying the number of chains to use in the Gibbs sampler (if method="BAYES"). More chains will improve the sensitivity of the convergence diagnostic, but will cause the simulation to run more slowly. The default number of chains is 4.
pars	A list with additional arguments. See 'Details' for more information. The following parameters configure the MCMC sampling procedure: hp.mu.mean (mean of the prior distribution of the random effects model, defaults to 0), hp.mu.var (variance of the prior distribution of the random effects model, defaults to 1E6), hp.tau.min (minimum value for the between-study standard deviation, defaults to 0), hp.tau.max (maximum value for the between-study standard deviation, defaults to 2), hp.tau.sigma (standard deviation of the prior distribution for the between-study standard-deviation), hp.tau.dist (prior distribution for the between-study standard-deviation. Defaults to "dunif"), hp.tau.df (degrees of freedom for the prior distribution for the between-study standard-deviation. Defaults to 3). Other arguments are method.restore.c.se (method for restoring missing estimates for the standard error of the c-statistic. See ccalc for more information), model.cstat (The likelihood/link for modeling the c-statistic; see "Details"), model.oe (The likelihood/link for modeling the O:E ratio; see "Details")
...	Additional arguments that are passed to **rma** or **runjags** (if method="BAYES").

Details

Meta-analysis of the concordance statistic: A summary estimate for the concorcance (c-) statistic can be obtained by specifying measure="cstat". The c-statistic is a measure of discrimination, and indicates the ability of a prediction model to distinguish between patients developing

and not developing the outcome. The c-statistic typically ranges from 0.5 (no discriminative ability) to 1 (perfect discriminative ability). When missing, the c-statistic and/or its standard error are derived from other reported information. See `ccalc` for more information.

By default, it is assumed that the logit of the c-statistic is Normally distributed within and across studies (`pars$model.cstat = "normal/logit"`). Alternatively, it is possible to assume that the raw c-statistic is Normally distributed across studies `pars$model.cstat = "normal/identity"`.

Meta-analysis of the total observed versus expected ratio: A summary estimate for the total observed versus expected (O:E) ratio can be obtained by specifying `measure="OE"`. The total O:E ratio provides a rough indication of the overall model calibration (across the entire range of predicted risks). When missing, the total O:E ratio and/or its standard error are derived from other reported information. See `oecalc` for more information.

For frequentist meta-analysis, within-study variation can either be modeled using a Normal (`model.oe = "normal/log"` or `model.oe = "normal/identity"`) or a Poisson distribution (`model.oe = "normal/log"`).

When performing a Bayesian meta-analysis, all data are modeled using a one-stage random effects (hierarchical related regression) model. In particular, a binomial distribution (if `O`, `E` and `N` is known), a Poisson distribution (if only `O` and `E` are known) or a Normal distribution (if `OE` and `OE.se` or `OE.95CI` are known) is selected separately for each study.

Bayesian meta-analysis: All Bayesian meta-analysis models assume random effects by default. Results are based on the posterior median. Credibility and prediction intervals are directly obtained from the corresponding posterior quantiles.

The prior distribution for the (transformed) summary estimate is always modeled using a Normal distribution, with mean `hp.mu.mean` (defaults to 0) and variance `hp.mu.var` (defaults to 1E6). For meta-analysis of the total O:E ratio, the maximum value for `hp.mu.var` is 100.

By default, the prior distribution for the between-study standard deviation is modeled using a uniform distribution (`hp.tau.dist="dunif"`), with boundaries `hp.tau.min` and `hp.tau.max`. Alternatively, it is possible to specify a truncated Student-t distribution (`hp.tau.dist="dhalft"`) with a mean of `hp.tau.mean`, a standard deviation of `hp.tau.sigma` and `hp.tau.df` degrees of freedom. This distribution is again restricted to the range `hp.tau.min` to `hp.tau.max`.

Value

An object of class `valmeta` with the following elements:

"data" array with (transformed) data used for meta-analysis, and method(s) used for restoring missing information.

"measure" character string specifying the performance measure that has been meta-analysed.

"method" character string specifying the meta-analysis method.

"model" character string specifying the meta-analysis model (link function).

"est" summary estimate for the performance statistic. For Bayesian meta-analysis, the posterior median is returned.

"ci.lb" lower bound of the confidence (or credibility) interval of the summary performance estimate.

"ci.ub" upper bound of the confidence (or credibility) interval of the summary performance estimate.

"pi.lb" lower bound of the (approximate) prediction interval of the summary performance estimate.

"pi.ub" upper bound of the (approximate) prediction interval of the summary performance estimate.

"fit" the full results from the fitted model. Only defined if `ret.fit = TRUE`.

"slab" vector specifying the label of each study.

Note

The width of calculated confidence, credibility and prediction intervals can be specified using `level` in the `pars` argument (defaults to 0.95).

References

Debray TPA, Damen JAAG, Snell KIE, Ensor J, Hooft L, Reitsma JB, et al. A guide to systematic review and meta-analysis of prediction model performance. BMJ. 2017;356:i6460.

Debray TPA, Damen JAAG, Riley R, Snell KIE, Reitsma JB, Hooft L, et al. A framework for meta-analysis of prediction model studies with binary and time-to-event outcomes. Stat Methods Med Res. 2018; In press.

Viechtbauer W. Conducting Meta-Analyses in R with the metafor Package. *Journal of Statistical Software*. 2010; 36(3). Available from: http://www.jstatsoft.org/v36/i03/

See Also

`ccalc` to calculate concordance statistics and corresponding standard errors, `oecalc` to calculate the total O:E ratio and corresponding standard errors, `plot.valmeta` to generate forest plots

Examples

```
######### Validation of prediction models with a binary outcome #########
data(EuroSCORE)

# Meta-analysis of the c-statistic (random effects)
fit <- with(EuroSCORE, valmeta(cstat=c.index, cstat.se=se.c.index,
                               cstat.95CI=cbind(c.index.95CIl,c.index.95CIu),
                               N=n, O=n.events, slab=Study))
plot(fit)

# Nearly identical results when we need to estimate the SE
with(EuroSCORE, valmeta(cstat=c.index,  N=n, O=n.events, slab=Study))

# Two-stage meta-analysis of the total O:E ratio (random effects)
with(EuroSCORE, valmeta(measure="OE", O=n.events, E=e.events, N=n))
with(EuroSCORE, valmeta(measure="OE", O=n.events, E=e.events))
with(EuroSCORE, valmeta(measure="OE", Po=Po, Pe=Pe, N=n))

# One-stage meta-analysis of the total O:E ratio (random effects)
with(EuroSCORE, valmeta(measure="OE", O=n.events, E=e.events, method="ML",
                        pars=list(model.oe="poisson/log")))
```

```
## Not run:
# Bayesian random effects meta-analysis of the c-statistic
fit2 <- with(EuroSCORE, valmeta(cstat=c.index, cstat.se=se.c.index,
                                cstat.95CI=cbind(c.index.95CIl,c.index.95CIu),
                                N=n, O=n.events, method="BAYES", slab=Study))

# Bayesian one-stage random effects meta-analysis of the total O:E ratio
# Consider that some (but not all) studies do not provide information on N
# A Poisson distribution will be used for studies 1, 2, 5, 10 and 20
# A Binomial distribution will be used for the remaining studies
EuroSCORE.new <- EuroSCORE
EuroSCORE.new$n[c(1, 2, 5, 10, 20)] <-  NA
pars <- list(hp.tau.dist="dhalft",   # Prior for the between-study standard deviation
             hp.tau.sigma=1.5,       # Standard deviation for 'hp.tau.dist'
             hp.tau.df=3,            # Degrees of freedom for 'hp.tau.dist'
           hp.tau.max=10)        # Maximum value for the between-study standard deviation
fit3 <- with(EuroSCORE.new, valmeta(measure="OE", O=n.events, E=e.events, N=n,
        method="BAYES", slab=Study, pars=pars, ret.fit = T))
plot(fit3)
print(fit3$fit$model) # Inspect the JAGS model
print(fit3$fit$data)  # Inspect the JAGS data

## End(Not run)

######### Validation of prediction models with a time-to-event outcome #########
data(Framingham)

# Meta-analysis of total O:E ratio after 10 years of follow-up
with(Framingham, valmeta(measure="OE", Po=Po, Pe=Pe, N=n))
```

`vcov.riley` *Calculate Variance-Covariance Matrix for a Fitted Riley Model Object*

Description

Returns the variance-covariance matrix of the main parameters of a fitted model object.

Usage

```
## S3 method for class 'riley'
vcov(object, ...)
```

Arguments

`object`	a `riley` object.
`...`	arguments to be passed on to other functions

Details

The variance-covariance matrix is obtained from the inverse Hessian as provided by `optim`.

Value

A matrix of the estimated covariances between the parameter estimates in the Riley model: logit of sensitivity (`mu1`), logit of false positive rate (`mu2`), additional variation of `mu1` beyond sampling error (`psi1`), additional variation of `mu2` beyond sampling error (`psi2`) and a transformation of the correlation between `psi1` and `psi2` (`rhoT`). The original correlation is given as `inv.logit(rhoT)*2-1`.

Note

A warning message is casted when the Hessian matrix contains negative eigenvalues. This implies that the identified minimum for the (restricted) negative log-likelihood is a saddle point, and that the solution is therefore not optimal.

Author(s)

Thomas Debray <thomas.debray@gmail.com>

References

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**, 172–186.

See Also

`riley`

Zhang — *Meta-analysis of the prognostic role of hormone receptors in endometrial cancer*

Description

This dataset comprises the results from 16 studies assessing the prognostic role of human epidermal growth factor receptor 2 (HER2) in endometrial cancer. These studies were previously identified in a systematic review by Zhang et al. to evaluate the overall risk of several hormone receptors for endometrial cancer survival.

Usage

```
data("Zhang")
```

Format

A data frame with 20 observations on the following 10 variables.

`Study` a factor with 16 levels to indicate the study

`PrimaryAuthor` a factor indicating the first author's last name

`year` a numeric vector indicating the publication year

`Country` a factor indicating the source country of the study data

`Disease` a factor indicating the studied disease. Possible levels are `EC` (endometrial cancer), `EEC` (endometrioid endometrial cancer) and `UPSC` (uterine papillary serous carcinoma)

`N` a numeric vector describing the total sample size of each study

`HR` a numeric vector describing the estimated hazard ratio of each study

`HR.025` a numeric vector describing the lower boundary of the 95% confidence interval of `HR`

`HR.975` a numeric vector describing the upper boundary of the 95% confidence interval of `HR`

`outcome` a factor indicating the studied outcome. Possible levels are `OS` (overall survival) and `PFS` (progression-free survival)

Details

Eligible studies were identified by searching the PubMed and EMBASE databases for publications from 1979 to May 2014. Data were collected from studies comparing overall survival or progression-free survival in patients with elevated levels of human epidermal growth factor receptor 2 with those in patients with lower levels.

Source

Zhang Y, Zhao D, Gong C, Zhang F, He J, Zhang W, et al. Prognostic role of hormone receptors in endometrial cancer: a systematic review and meta-analysis. World J Surg Oncol. 2015 Jun 25;13:208.

References

Riley RD, Jackson D, Salanti G, Burke DL, Price M, Kirkham J, et al. Multivariate and network meta-analysis of multiple outcomes and multiple treatments: rationale, concepts, and examples. BMJ. 2017 13;358:j3932.

Examples

```
data(Zhang)

# Display the hazard ratios for overall survival in a forest plot
ds <- subset(Zhang, outcome=="OS")
with(ds, forest(theta = HR, theta.ci.lb = HR.025, theta.ci.ub = HR.975,
               theta.slab = Study, xlab = "Hazard ratio of HER2 versus OS", refline = 1))
```

Index