Package 'metasens'

September 29, 2020

Title Advanced Statistical Methods to Model and Adjust for Bias in Meta-Analysis

Version 0.5-0

Date 2020-09-29

Depends meta (>= 4.14-0)

URL https://github.com/guido-s/metasens

http://meta-analysis-with-r.org

Description The following methods are implemented to evaluate how sensitive the results of a meta-analysis are to potential bias in meta-analysis and to support Schwarzer et al. (2015) <DOI:10.1007/978-3-319-21416-0>, Chapter 5 'Small-Study Effects in Meta-Analysis':
- Copas selection model described in Copas & Shi (2001) <DOI:10.1177/096228020101000402>;
- limit meta-analysis by Rücker et al. (2011) <DOI:10.1093/biostatistics/kxq046>;
- upper bound for outcome reporting bias by Copas & Jackson (2004) <DOI:10.1111/j.0006-341X.2004.00161.x>;
- imputation methods for missing binary data by Gamble & Hollis (2005) <DOI:10.1016/j.jclinepi.2004.09.013> and Higgins et al. (2008) <DOI:10.1177/1740774508091600>.

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.1.1

NeedsCompilation no

Author Guido Schwarzer [cre, aut] (<https://orcid.org/0000-0001-6214-9087>), James R. Carpenter [aut] (<https://orcid.org/0000-0003-3890-6206>), Gerta Rücker [aut] (<https://orcid.org/0000-0002-2192-2560>)

Maintainer Guido Schwarzer <sc@imbi.uni-freiburg.de>

Repository CRAN

Date/Publication 2020-09-29 19:30:02 UTC

R topics documented:

`metasens-package` *metasens: Brief overview of methods and general hints*

Description

R package **metasens** provides advanced statistical methods to model and adjust bias in meta-analysis and supports Schwarzer et al. (2015), Chapter 5 "Small-Study Effects in Meta-Analysis" http://meta-analysis-with-r.org/.

Details

R package **metasens** is an add-on package for **meta** providing the following meta-analysis methods:

- Copas selection model (function `copas`) described in Copas & Shi (2001) and evaluated in Schwarzer et al., 2010);
- limit meta-analysis (`limitmeta`) by Rücker et al. (2011);
- upper bound for outcome reporting bias (`orbbound`) described in Copas & Jackson (2004);
- imputation methods for missing binary data (`metamiss`) described in Gamble & Hollis (2005) and Higgins et al. (2008).

Furthermore, functions and datasets from **metasens** are utilised in Schwarzer et al. (2015), Chapter 5 "Small-Study Effects in Meta-Analysis", http://meta-analysis-with-r.org/.

Type `help(package = "metasens")` for a listing of R functions available in **metasens**.

Type `citation("metasens")` on how to cite **metasens** in publications.

To report problems and bugs

- type `bug.report(package = "metasens")` if you do not use RStudio,
- send an email to Guido Schwarzer <sc@imbi.uni-freiburg.de> if you use RStudio.

The development version of **metasens** is available on GitHub https://github.com/guido-s/metasens.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

References

Copas J, Jackson D (2004): A bound for publication bias based on the fraction of unpublished studies. *Biometrics*, **60**, 146–53

Copas JB, Shi JQ (2001): A sensitivity analysis for publication bias in systematic reviews. *Statistical Methods in Medical Research*, **10**, 251–65

Gamble C, Hollis S (2005): Uncertainty method improved on best–worst case analysis in a binary meta-analysis. *Journal of Clinical Epidemiology*, **58**, 579–88

Higgins JPT, White IR, Wood AM (2008): Imputation methods for missing outcome data in meta-analysis of clinical trials. *Clinical Trials*, **5**, 225–39

Rücker G, Schwarzer G, Carpenter JR, Binder H, Schumacher M (2011): Treatment-effect estimates adjusted for small-study effects via a limit meta-analysis. *Biostatistics*, **12**, 122–42

Schwarzer G, Carpenter J, Rücker G (2010): Empirical evaluation suggests Copas selection model preferable to trim-and-fill method for selection bias in meta-analysis. *Journal of Clinical Epidemiology*, **63**, 282–8

Schwarzer G, Carpenter JR, Rücker G (2015): *Meta-Analysis with R (Use-R!)*. Springer International Publishing, Switzerland

---

`copas` *Copas selection model analysis*

Description

Perform a Copas selection model analysis for selection bias in meta-analysis.

Usage

```
copas(
  x,
  level.comb = x$level.comb,
  gamma0.range = NULL,
  gamma1.range = NULL,
  ngrid = 20,
  nlevels = 10,
  levels = NULL,
```

```
  slope = NULL,
  left = NULL,
  rho.bound = 0.9999,
  sign.rsb = 0.1,
  backtransf = x$backtransf,
  title = x$title,
  complab = x$complab,
  outclab = x$outclab,
  silent = TRUE,
  warn = options()$warn
)
```

Arguments

x	An object of class meta, obtained from one of the functions metabin, metacont and metagen in the package meta.
level.comb	The level used to calculate confidence intervals for pooled estimates.
gamma0.range	(Advanced users only) A numerical vector of length two specifying the range of gamma0 values the program will explore. The parameter gamma0 is the constant in the probit selection model for study publication. Thus, the cumulative normal of gamma0 is approximately the probability that a small study is published (in non-technical terms gamma0 relates to the probability of publishing a small study, although its values are not restricted to the range [0,1]; larger values correspond to higher probabilities of publishing a small study). Most users will not need to specify a range for this parameter. When no argument is specified, the program uses an algorithm to determine a suitable range. This is based on the range of treatment effect standard errors in the meta-analysis, and is described in more detail below.
gamma1.range	(Advanced users only) A numerical vector of length two specifying the range of gamma1 values the program will explore. The parameter gamma1 is the coefficient of study precision (1/standard error) in the probit selection model for study publication (in non-technical terms gamma1 relates to the rate at which the probability of publishing a study increases as the standard error of the treatment effect it reports decreases; larger values correspond to higher probabilities of publishing a small study). Most users will not need to specify a range for this parameter. When no argument is specified, the program uses an algorithm to determine a suitable range. This is based on the range of treatment effect standard errors in the meta-analysis, and is described in more detail below.
ngrid	The program fits the Copas selection model over a grid defined by the range of values of gamma0 and gamma1 specified in the previous two arguments. This parameter fixes the square-root of the number of points in the grid.
nlevels	(Advanced users only). Fitting the Copas model over the grid specified by the previous three arguments results in a treatment estimate at every point in the grid. These can then be displayed on a contour plot where contours of treatment effect (z-axis) are shown by gamma0 (x-axis) and gamma1 (y-axis). This argument specifies the number of contour lines that will be drawn.

Note

(i) Calculations for the contour plot are performed by the function `copas`, so this argument has no effect in the `plot` function.

(ii) If a large number of contour lines are desired, then you may wish to consider increasing the grid size (argument `ngrid` above).

Leave this option unspecified if you are using the option `levels` below.

`levels` A numerical vector of treatment values for which contour lines will be drawn. In more detail, fitting the Copas model over the grid specified by the arguments `gamma0.range`, `gamma1.range` and `ngrid` results in a treatment estimate at every point in the grid. These are then displayed on a contour plot where contours of treatment effect (z-axis) are shown by gamma0 (x-axis) and gamma1 (y-axis). This argument is a numerical vector which specifies the treatment effects for which contour lines will be drawn.

It is usually not a good idea to set this argument for initial runs, as one does not know the range of treatment values that the contour plot will cover, and treatment values which do not correspond to values in the contour plot (defined by the range of gamma0 and gamma1) will not be plotted.

Note

(i) Calculations for the contour plot are performed by the function `copas`, so this argument has no effect in the `plot` function.

(ii) Contours will not be drawn if a large number of contour lines are desired, then you may wish to consider increasing the grid size (argument `ngrid` above).

Leave this option unspecified if you are using the option `nlevels` above.

`slope` A numeric providing the slope of the line approximately orthogonal to contours in the contour plot. If the argument `slope` is `NULL` (default) the program seeks to estimate the slope of the contours in the region of the maximum, which are usually approximately parallel. Most users will leave the argument `slope` unspecified, at least for the first analysis of a data set, but in certain cases setting it manually can improve the results.

`left` A logical indicating whether the cause of any selection bias is due to missing studies on the left or right of the funnel plot: left hand side if `left=TRUE`, right hand side if `left=FALSE`. This information is needed in order to be sure the test for presence of residual selection bias is calculated correctly. If not set, the linear regression test for funnel plot asymmetry (i.e., function `metabias(...,meth="linreg")`) is used to determine whether studies are missing on the left or right hand side. In the majority of cases this will work correctly.

`rho.bound` (Advanced users only) A number giving the upper bound for the correlation parameter `rho` (see details below). This must be < 1, and usually > 0.95. The lower bound is calculated as -(the upper bound).

`sign.rsb` The significance level for the test of residual selection bias (between 0 and 1).

`backtransf` A logical indicating whether results should be back transformed in printouts and plots. If `backtransf=TRUE` (default), results for `sm="OR"` are printed as odds ratios rather than log odds ratio, for example.

`title` Title of meta-analysis / systematic review.

`complab` Comparison label.

`outclab` Outcome label.

`silent` A logical indicating whether information on progress in fitting the Copas selection model should be printed: `silent=TRUE`, do not print information (the default); `silent=FALSE`, print information.

`warn` A number setting the handling of warning messages. It is not uncommon for numerical problems to be encountered during estimation over the grid of (gamma0, gamma1) values. Usually this does not indicate a serious problem. This option specifies what to do with warning messages. `warn=-1`: ignore all warnings; `warn=0` (the default): store warnings till function finishes; if there are less than 10, print them, otherwise print a message saying warning messages were generated; `warn=1`: print warnings as they occur; `warn=2`: stop the function when the first warning is generated. For further details see `help(options)`.

Details

The program takes an object of class `meta`, which is most easily created by an analysis using one of the functions `metabin`, `metacont` and `metagen` in the package meta, performs a 'Copas selection model analysis' and presents a graphical and tabular summary of the results. An object of class `copas` is created and this can be used to recreate the results table and graphs subsequently, without re-running the analysis, using the `print`, `summary` and `plot` function.

Conduct a Copas selection model analysis to investigate, and attempt to correct for, selection / publication bias in a meta-analysis.

The Copas selection model consists of two models, which are fitted jointly. The first is the usual random effects meta-analysis model, and the second is a selection model, where study i is selected for publication if Z>0, where

Z = gamma0 + gamma1 / (SE(i)) + delta(i)

The error delta(i) is correlated with the error in the random effects meta-analysis, with correlation rho. If rho=0, the model corresponds to the usual random effects meta-analysis. As rho moves from 0 to 1, studies with larger treatment estimates are more likely to be selected/published.

The software chooses a grid of gamma0 and gamma1 values, corresponding to a range of selection / publication probabilities for the study with the largest treatment effect standard error (often the smallest study). For each value in this grid, the treatment effect is estimated using the function `optim`. This information is used to produce the contour plot (top right panel of output from `plot.copas`).

Contours of constant treatment effect are usually locally parallel. The software estimates the slope of these contours, and combines this information with other parameter estimates from the model to explore (i) how the treatment estimate, and its standard error, change with increasing selection (bottom left panel, `plot.copas`) and (ii) how much selection needs to be accounted for before any remaining asymmetry in the funnel plot is likely to have occurred by chance (bottom right panel, `plot.copas`).

A table of results can be produced by the function `summary.copas`. A more detail output is provided by the function `print.copas`.

For a fuller description of the model, our implementation and specifically our approach to estimating the locally parallel contours, see Carpenter et al. (2009) and Schwarzer et al. (2010).

Value

An object of class copas with corresponding print, summary, and plot function. The object is a list containing the following components:

TE	Vector of treatment effects plotted in treatment effect plot
seTE	Vector of standard error of TE
TE.random	Usual random effects estimate of treatment effect
seTE.random	Standard error of TE.random
lower.random	Lower confidence limit of usual random effects estimate
upper.random	Upper confidence limit of usual random effects estimate
statistic.random	Test statistic of an overall effect (usual random effects model)
pval.random	P-value of test of overall effect (usual random effects model)
TE.adjust	Adjusted random effects estimate from Copas selection model
seTE.adjust	Standard error of TE.adjust
lower.adjust	Lower confidence limit of adjusted treatment estimate
upper.adjust	Upper confidence limit of adjusted treatment estimate
statistic.adjust	Test statistic of an overall effect (Copas selection model)
pval.adjust	P-value of test of overall effect (Copas selection model)
left	Whether selection bias expected on left or right
rho.bound	Bound on rho
gamma0.range	Range of gamma0 (see help on copas arguments above)
gamma1.range	Range of gamma1 (see help on copas arguments above)
slope	Slope of line approximately orthogonal to contours in contour plot
regr	A list containing information on regression lines fitted to contours in contour plot
ngrid	Square root of grid size
nlevels	Number of contour lines
gamma0	Vector of gamma0 values at which model fitted (determined by gamma0.range and grid). x-axis values for contour plot
gamma1	vector of gamma1 values at which model fitted (determined by gamma1.range and grid). y-axis values for contour plot
TE.contour	Treatment values (ie z-axis values) used to draw contour plot.
x.slope	x coordinates for 'orthogonal line' in contour plot
y.slope	y coordinates for 'orthogonal line' in contour plot
TE.slope	Vector of treatment values plotted in treatment effect plot
seTE.slope	Standard error of TE.slope
rho.slope	Vector of estimated rho values corresponding to treatment estimates in TE.slope

tau.slope	Vector of estimated heterogeneity values corresponding to treatment estimates in TE.slope
loglik1	Vector of log-likelihood values corresponding to treatment estimates in TE.slope
conv1	Numerical vector indicating convergence status for each treatment estimate in TE.slope - see parameter convergence in function optim
message1	Character vector - translation of conv1
loglik2	Vector of log-likelihoods from fitting model to evaluate presence of residual selection bias
conv2	Numerical vector indicating convergence status for models to evaluate presence of residual selection bias - see parameter convergence in function optim
message2	Character vector - translation of conv2
publprob	Vector of probabilities of publishing the smallest study, used in x-axis of bottom two panels in function plot.copas
pval.rsb	P-values for tests on presence of residual selection bias, plotted in bottom right panel in plot.copas
sign.rsb	The significance level for the test of residual selection bias
N.unpubl	Approximate number of studies the model suggests remain unpublished
sm	Effect measure (e.g., for binary data, OR - odds ratio, RR - risk ratio, RD - risk difference, AS - arcsin difference)
title	Title of meta-analysis / systematic review.
complab	Comparison label.
outclab	Outcome label.
call	Call to copas function
version	Version of R package metasens used to create object.
x	Details of meta-analysis object used as input into copas function

Author(s)

James Carpenter <James.Carpenter@lshtm.ac.uk>, Guido Schwarzer <sc@imbi.uni-freiburg.de>

References

Carpenter JR, Schwarzer G, Rücker G, Künstler R (2009): Empirical evaluation showed that the Copas selection model provided a useful summary in 80% of meta-analyses. *Journal of Clinical Epidemiology*, **62**, 624–31

Copas J (1999): What works?: Selectivity models and meta-analysis. *Journal of the Royal Statistical Society, Series A*, **162**, 95–109

Copas J, Shi JQ (2000): Meta-analysis, funnel plots and sensitivity analysis. *Biostatistics*, **1**, 247–62

Copas JB, Shi JQ (2001): A sensitivity analysis for publication bias in systematic reviews. *Statistical Methods in Medical Research*, **10**, 251–65

Schwarzer G, Carpenter J, Rücker G (2010): Empirical evaluation suggests Copas selection model preferable to trim-and-fill method for selection bias in meta-analysis. *Journal of Clinical Epidemiology*, **63**, 282–8

See Also

plot.copas, summary.copas, metabias, metagen, funnel

Examples

```
data(Fleiss1993bin, package = "meta")

# Perform meta-analysis
#  (Note d.asp indicates deaths, n.asp total in aspirin group;
#        d.plac indicates deaths, n.plac total in placebo group)
#
m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")
summary(m1)

# Perform a basic Copas selection model analysis
#
cop1 <- copas(m1)
plot(cop1)
summary(cop1)
#
# Interpretation:
#
# a. The initial meta-analysis shows the fixed and random effects
#    pooled ORs differ; consistent with asymmetry in the funnel
#    plot and possible selection bias. Both fixed effect and random
#    effects model show a significant treatment effect in this
#    dataset.
#
# b. Plotting the copas analysis shows
#
# (i) funnel plot: asymmetry indicates possible selection bias.
#
# (ii) contour plot treatment effect declines steadily as selection
#      increases (no selection, top right, log OR < -0.12;
#      increasing selection as move to left of plot, log OR rises
#      to -0.03.
#

# (iii) Treatment effect plot suggests that even with no selection,
#       p-value for treatment effect is larger than 0.05 which is
#       different from the result of the usual random effects model
#       (see output of summary(cop1). This difference is due to the
#       use of different methods to estimate the between-study
#       variance: maximum-likelihood in Copas analysis compared to
#       method-of-moments in usual random effects model.  The
#       p-value for treatment effect is increasing with increasing
#       selection.
#
# (iv) P-value for residual selection bias plot: this shows that
#      even with no selection bias, the p-value for residual
#      selection bias is non-significant at the 10% level. As
#      expected, as selection increases the p-value for residual
```

```
#      selection bias increases too.


# Repeat the same example, setting several arguments of the copas
# function:
#
cop2 <- copas(m1,
              gamma0.range = c(-0.5, 2.1), # range of gamma0 parameter
              gamma1.range = c(0, 0.08),   # range of gamma1 parameter
              ngrid = 20,                  # specify a 20x20 grid (finer than default)
              levels = c(-0.13, -0.12, -0.1, -0.09,
                         -0.07, -0.05, -0.03), # specify contour lines
              slope = 0.2,       # specify slope of 'orthogonal' line in contour plot
              left = FALSE,      # as any selection bias due to missing studies on right
              rho.bound = 0.998, # constrain rho between [-0.998, 0.998]
              silent = FALSE,    # update user on progress
              warn = -1          # suppress warning messages
             )
plot(cop2)
#
# Print table of results used to draw treatment effect plot:
#
summary(cop2)
```

`Crowther2003` *Aspirin after Myocardial Infarction*

Description

Meta-analysis on phenobarbital prior to preterm birth for preventing neonatal periventricular haemorrhage

Format

A data frame with the following columns:

study	study label
pvh.e	number of periventricular haemorrhages in experimental group
n.e	number of observations in experimental group
pvh.c	number of periventricular haemorrhages in control group
n.c	number of observations in control group

Source

Crowther CA, Henderson-Smart DJ (2003): Phenobarbital prior to preterm birth for preventing neonatal periventricular haemorrhage. *Cochrane Database of Systematic Reviews*, CD000164

Examples

```
data(Crowther2003)
metabin(pvh.e, n.e, pvh.c, n.c, data = Crowther2003, studlab = study)
```

`forest.orbbound` *Forest plot for* `orbbound` *object (bound for outcome reporting bias)*

Description

Draws a forest plot in the active graphics window (using grid graphics system).

Usage

```
## S3 method for class 'orbbound'
forest(
  x,
  comb.fixed = x$x$comb.fixed,
  comb.random = x$x$comb.random,
  text.fixed = "FE model",
  text.random = "RE model",
  smlab = NULL,
  leftcols = c("studlab", "maxbias"),
  leftlabs = c("Missing\nstudies", "Maximum\nbias"),
  backtransf = x$backtransf,
  digits = max(3, .Options$digits - 3),
  ...
)
```

Arguments

`x`	An object of class `orbbound`.
`comb.fixed`	A logical indicating whether sensitivity analysis for fixed effect model should be plotted.
`comb.random`	A logical indicating whether sensitivity analysis for random effects model should be plotted.
`text.fixed`	A character string used in the plot to label subgroup with results for fixed effect model.
`text.random`	A character string used in the plot to label subgroup with results for random effects model.
`smlab`	A label printed at top of figure. If only results for either fixed effect or random effects model is plotted, text indicates which model was used.
`leftcols`	A character vector specifying (additional) columns to be plotted on the left side of the forest plot or a logical value (see `forest.meta` help page for details).
`leftlabs`	A character vector specifying labels for (additional) columns on left side of the forest plot (see `forest.meta` help page for details).

backtransf	A logical indicating whether results should be back transformed in printouts and plots. If backtransf=TRUE (default), results for sm="OR" are printed as odds ratios rather than log odds ratio, for example.
digits	Minimal number of significant digits, see print.default.
...	Additional arguments for forest.meta function.

Details

A forest plot, also called confidence interval plot, is drawn in the active graphics window.

For relative effect measures, e.g., 'RR', 'OR', and 'HR', the column labeled "Maximum bias" contains the relative bias, e.g. a value of 1.10 means a maximum overestimation by 10 percent. If backtransf=FALSE for these summary measures, maximum bias is instead printed as absolute bias.

Internally, R function forest.meta is called to create a forest plot. For more information see help page of the forest.meta function.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

orbbound, print.orbbound

Examples

```
data(Fleiss1993bin, package = "meta")

m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")

orb1 <- orbbound(m1, k.suspect = 1:5)
print(orb1, digits = 2)
forest(orb1, xlim = c(0.7, 1.5))
## Not run: forest(orb1, backtransf = FALSE)
```

---

funnel.limitmeta *Funnel plot for limit meta-analysis*

Description

Draws a funnel plot in the active graphics window.

Usage

```
## S3 method for class 'limitmeta'
funnel(
  x,
  pch = 21,
  cex = 1,
  col = "black",
  bg = "darkgray",
  lwd = 1,
  pch.adjust = 18,
  cex.adjust = 1.5,
  col.adjust = "gray",
  bg.adjust = "gray",
  line = TRUE,
  xmin.line,
  xmax.line,
  lty.line = 1,
  lwd.line = lwd,
  col.line = "gray",
  shrunken = FALSE,
  pch.shrunken = 22,
  cex.shrunken = 1,
  col.shrunken = "black",
  bg.shrunken = "white",
  lty.connect = 1,
  lwd.connect = 0.8,
  col.connect = "black",
  backtransf = x$backtransf,
  ...
)
```

Arguments

x	An object of class limitmeta.
pch	The plotting symbol used for individual studies.
cex	The magnification to be used for plotting symbol.
col	A vector with colour of plotting symbols.
bg	A vector with background colour of plotting symbols (only used if pch in 21:25).
lwd	The line width for confidence intervals (see funnel).
pch.adjust	The plotting symbol used for the adjusted effect estimate.
cex.adjust	The magnification to be used for the plotting symbol of the adjusted effect estimate.
col.adjust	Colour of plotting symbol for adjusted effect estimate.
bg.adjust	Background colour of plotting symbol for adjusted effect estimate.
line	A logical indicating whether adjusted regression line should be plotted.

xmin.line	Minimal value for the adjusted regression line (on x-axis).
xmax.line	Maximum value for the adjusted regression line (on x-axis).
lty.line	Line type of the adjusted regression line.
lwd.line	The line width of the adjusted regression line.
col.line	Color of the adjusted regression line.
shrunken	A logical indicating whether shrunken treatment estimates should be plotted.
pch.shrunken	The plotting symbol used for shrunken effect estimates.
cex.shrunken	The magnification to be used for the plotting symbol of the shrunken effect estimates.
col.shrunken	Colour of plotting symbol for shrunken effect estimates.
bg.shrunken	Background colour of plotting symbol for shrunken effect estimates.
lty.connect	Line type for line connecting original and shrunken treatment estimates.
lwd.connect	The line width of the connecting lines.
col.connect	Color of the connecting lines.
backtransf	A logical indicating whether results should be back transformed in printouts and plots. If backtransf=TRUE (default), results for sm="OR" are printed as odds ratios rather than log odds ratio, for example.
...	Additional arguments for funnel function.

Details

A funnel plot is drawn in the active graphics window. In addition this function adds the adjusted effect estimate as well as a nonlinear regression line (also called adjusted regression line) if argument line is TRUE. The adjusted regression line is representing the dependence of the treatment effect estimate on the standard error across studies. The adjusted regression line is only plotted in addition to the adjusted treatment effect if argument method.adjust="beta0" (default) has been used in the limitmeta function.

If argument shrunken is TRUE the shrunken effect estimates are also plotted. Lines are connecting original and shrunken effect estimates.

Internally, R function funnel.meta is called to create a funnel plot. For more information see help page of the funnel function.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>, Gerta Rücker <ruecker@imbi.uni-freiburg.de>

See Also

limitmeta, funnel

Examples

```
data(Moore1998)
m1 <- metabin(succ.e, nobs.e, succ.c, nobs.c,
              data = Moore1998, sm = "OR", method = "Inverse")

print(summary(limitmeta(m1)), digits = 2)
funnel(limitmeta(m1))

# Print results on log scale
#
print(summary(limitmeta(m1)), digits = 2, backtransf = FALSE)
funnel(limitmeta(m1), backtransf = FALSE)
```

`limitmeta` *Limit meta-analysis*

Description

Implementation of the limit meta-analysis method by Rücker et al. (2011) to adjust for bias in meta-analysis.

Usage

```
limitmeta(
  x,
  method.adjust = "beta0",
  level = x$level,
  level.comb = x$level.comb,
  backtransf = x$backtransf,
  title = x$title,
  complab = x$complab,
  outclab = x$outclab
)
```

Arguments

`x`	An object of class `meta`.
`method.adjust`	A character string indicating which adjustment method is to be used. One of `"beta0"`, `"betalim"`, or `"mulim"`, can be abbreviated.
`level`	The level used to calculate confidence intervals for individual studies.
`level.comb`	The level used to calculate confidence intervals for pooled estimates.
`backtransf`	A logical indicating whether results should be back transformed in printouts and plots. If `backtransf=FALSE`, results for the odds ratio are printed as log odds ratios rather than odds ratio, for example.
`title`	Title of meta-analysis / systematic review.
`complab`	Comparison label.
`outclab`	Outcome label.

Details

This function provides the method by Rücker et al. (2011) to estimate an effect estimate adjusted for bias in meta-analysis. The underlying model is an extended random effects model that takes account of possible small study effects by allowing the treatment effect to depend on the standard error:

theta(i) = beta + sqrt(SE(i)^2 + tau^2)(epsilon(i) + alpha),

where epsilon(i) follows a standard normal distribution. Here theta(i) is the observed effect in study i, beta the global mean, SE(i) the within-study standard error, and tau^2 the between-study variance. The parameter alpha represents the bias introduced by small-study effects. On the one hand, alpha can be interpreted as the expected shift in the standardized treatment effect if precision is very small. On the other hand, theta(adj) = beta + tau*alpha is interpreted as the limit treatment effect for a study with infinite precision (corresponding to SE(i) = 0).

Note that as alpha is included in the model equation, beta has a different interpretation as in the usual random effects model. The two models agree only if alpha=0. If there are genuine small-study effects, the model includes a component making the treatment effect depend on the standard error. The expected treatment effect of a study of infinite precision, beta + tau*alpha, is used as an adjusted treatment effect estimate.

The maximum likelihood estimates for alpha and beta can be interpreted as intercept and slope in linear regression on a so-called generalised radial plot, where the x-axis represents the inverse of sqrt(SE(i)^2 + tau^2) and the y-axis represents the treatment effect estimates, divided by sqrt(SE(i)^2 + tau^2).

Two further adjustments are available that use a shrinkage procedure. Based on the extended random effects model, a limit meta-analysis is defined by inflating the precision of each study with a common factor. The limit meta-analysis yields shrunken estimates of the study-specific effects, comparable to empirical Bayes estimates. Based on the extended random effects model, we obtain three different treatment effect estimates that are adjusted for small-study effects:

- an estimate based on the expectation of the extended random effects model, beta0 = beta + tau*alpha (`method.adjust="beta0"`)
- the extended random effects model estimate of the limit meta-analysis, including bias parameter (`method.adjust="betalim"`)
- the usual random effects model estimate of the limit meta-analysis, excluding bias parameter (`method.adjust="mulim"`)

See Rücker, Schwarzer et al. (2011), Section 7, for the definition of G^2 and the three heterogeneity statisticics `Q`, `Q.small`, and `Q.resid`.

For comparison, the original random effects meta-analysis is always printed in the sensitivity analysis.

Value

An object of class `"limitmeta"` with corresponding `print`, `summary` and `funnel` function. The object is a list containing the following components:

`x, level, level.comb,method.adjust,title, complab, outclab`
: As defined above.

`TE, seTE` Estimated treatment effect and standard error of individual studies.

`TE.limit, seTE.limit`
Shrunken estimates and standard error of individual studies.

`studlab` Study labels.

`TE.random, seTE.random`
Unadjusted overall treatment effect and standard error (random effects model).

`lower.random, upper.random`
Lower and upper confidence interval limits (random effects model).

`statistic.random, pval.random`
Statistic and corresponding p-value for test of overall treatment effect (random effects model).

`w.random` Weight of individual studies (in random effects model).

`tau` Square-root of between-study variance.

`TE.adjust, seTE.adjust`
Adjusted overall effect and standard error (random effects model).

`lower.adjust, upper.adjust`
Lower and upper confidence interval limits for adjusted effect estimate (random effects model).

`statistic.adjust, pval.adjust`
Statistic and corresponding p-value for test of overall treatment effect for adjusted estimate (random effects model).

`alpha.r` Intercept of the linear regression line on the generalised radial plot, here interpreted as bias parameter in an extended random effects model. Represents the expected shift in the standardized treatment effect if precision is very small.

`beta.r` Slope of the linear regression line on the generalised radial plot.

`Q` Heterogeneity statistic.

`Q.small` Heterogeneity statistic for small study effects.

`Q.resid` Heterogeneity statistic for residual heterogeneity beyond small study effects.

`G.squared` Heterogeneity statistic G^2 (ranges from 0 to 100%).

`k` Number of studies combined in meta-analysis.

`call` Function call.

`version` Version of R package metasens used to create object.

Author(s)

Gerta Rücker <ruecker@imbi.uni-freiburg.de>, Guido Schwarzer <sc@imbi.uni-freiburg.de>

References

Rücker G, Carpenter JR, Schwarzer G (2011): Detecting and adjusting for small-study effects in meta-analysis. *Biometrical Journal*, **53**, 351–68

Rücker G, Schwarzer G, Carpenter JR, Binder H, Schumacher M (2011): Treatment-effect estimates adjusted for small-study effects via a limit meta-analysis. *Biostatistics*, **12**, 122–42

See Also

funnel.limitmeta, print.limitmeta

Examples

```
data(Moore1998)
m1 <- metabin(succ.e, nobs.e, succ.c, nobs.c,
              data = Moore1998, sm = "OR", method = "Inverse")

print(summary(limitmeta(m1)), digits = 2)
```

---

metamiss *Imputation methods for missing binary data*

Description

Imputation methods for the meta-analysis of binary outcomes with missing data.

Usage

```
metamiss(
  x,
  miss.e,
  miss.c,
  IMOR.e,
  IMOR.c = IMOR.e,
  method.miss = if (missing(IMOR.e)) "0" else "IMOR",
  small.values = "good",
  comb.fixed = x$comb.fixed,
  comb.random = x$comb.random,
  prediction = x$prediction
)
```

Arguments

x	An object of class metabin.
miss.e	Number of missing observations in experimental group.
miss.c	Number of missing observations in control group.
IMOR.e	IMOR in experimental group (see Details).
IMOR.c	IMOR in control group (see Details).
method.miss	A character string indicating which method is used to impute missing values. Either "GH", "IMOR", "0", "1", "pc", "pe", "p", "b", or "w", can be abbreviated (see Details).
small.values	A character string specifying whether small treatment effects indicate a beneficial ("good") or harmful ("bad") effect, can be abbreviated (see Details).

comb.fixed	A logical indicating whether a fixed effect meta-analysis should be conducted.
comb.random	A logical indicating whether a random effects meta-analysis should be conducted.
prediction	A logical indicating whether a prediction interval should be printed.

Details

This function provides several imputation methods to deal with missing data in the meta-analysis of binary outcomes (Gamble & Hollis, 2005; Higgins et al., 2008). In order to utilise these methods, the number of observations with missing outcomes must be provided for the experimental and control group (arguments miss.e and miss.c).

The following imputation methods for missing binary data are available.

Argument	Method
method.miss = "GH"	Method by Gamble & Hollis (2005)
method.miss = "IMOR"	Based on group-specific IMORs
method.miss = "0"	Imputed as no events, (i.e., 0)
method.miss = "1"	Imputed as events (i.e., 1)
method.miss = "pc"	Based on observed risk in control group
method.miss = "pe"	Based on observed risk in experimental group
method.miss = "p"	Based on group-specific risks
method.miss = "b"	Best case scenario for experimental group
method.miss = "w"	Worst case scenario for experimental group

The method by Gamble & Hollis (2005) is based on uncertainty intervals for individual studies resulting from best and worst case scenarios taking the missing data into account. The uncertainty intervals are used to calculate (inflated) standard errors which are considered in a generic inverse variance meta-analysis instead of the standard errors from the complete case meta-analysis.

All other methods are based on the Informative Missingness Odds Ratio (IMOR) which is defined as the odds of an event in the missing group over the odds of an event in the observed group (Higgins et al., 2008). For example, an IMOR of 2 means that the odds for an event is assumed to be twice as likely for missing observations. For method.miss = "IMOR", the IMORs in the experimental (argument IMOR.e) and control group (argument IMOR.c) must be specified by the user. For all other methods, the input for arguments IMOR.e and IMOR.c is ignored as these values are determined by the respective imputation method (see Table 2 in Higgins et al., 2008).

For the best and worst case scenarios (i.e., argument method.miss equal to "b" or "w"), the user has to specify whether the outcome is beneficial (argument small.values = "good") or harmful (small.values = "bad").

Value

An object of class c("metamiss","metagen","meta") with corresponding print, summary, and forest functions. See metagen for more information.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

References

Gamble C, Hollis S (2005): Uncertainty method improved on best–worst case analysis in a binary meta-analysis. *Journal of Clinical Epidemiology*, **58**, 579–88

Higgins JPT, White IR, Wood AM (2008): Imputation methods for missing outcome data in meta-analysis of clinical trials. *Clinical Trials*, **5**, 225–39

See Also

metabin, metagen

Examples

```
d1 <- data.frame(author = c("Beasley", "Selman"),
                 resp.h = c(29, 17), fail.h = c(18, 1), drop.h = c(22, 11),
                 resp.p = c(20, 7), fail.p = c(14, 4), drop.p = c(34, 18))
m1 <- metabin(resp.h, resp.h + fail.h, resp.p, resp.p + fail.p,
              data = d1, studlab = author, sm = "RR", method = "I")
m1

# Treat missings as no events
metamiss(m1, drop.h, drop.p)

# Assume IMORs of 2 for both experimental and control group
metamiss(m1, drop.h, drop.p, IMOR.e = 2)

# Gamble & Hollis (2005)
d2 <- data.frame(author = c("Lefevre", "van Vugt", "van Vugt"),
                 year = c(2001, 2000, 1998),
                 para.al = c(7, 4, 49), n.al = c(155, 134, 273),
                 miss.al = c(9, 16, 36),
                 para.ma = c(0, 0, 7), n.ma = c(53, 47, 264),
                 miss.ma = c(2, 3, 44))

m2 <- metabin(para.al, n.al, para.ma, n.ma,
              data = d2, studlab = paste0(author, " (", year, ")"),
              method = "Inverse", method.tau = "DL",
              sm = "OR")

metamiss(m2, miss.al, miss.ma, method = "GH")
```

Moore1998 *NSAIDS in acute pain*

Description

Meta-analysis on the effectiveness of topical non-steroidal anti-inflammatory drugs (NSAIDS) in acute pain.

Treatment success is defined as a reduction in pain of at least 50%.

Format

A data frame with the following columns:

study	study number
succ.e	number of treatment successes in NSAIDS group
nobs.e	number of patients in NSAIDS group
succ.c	number of treatment successes in control group
nobs.c	number of patients in control group

Source

Moore RA, Tramer MR, Carroll D, Wiffen PJ, McQuay HJ (1998): Quantitive systematic review of topically applied non-steroidal anti-inflammatory drugs. *British Medical Journal*, **316**, 333–8

Examples

```
data(Moore1998)
m1 <- metabin(succ.e, nobs.e, succ.c, nobs.c,
              data = Moore1998, sm = "OR", method = "Inverse")

print(limitmeta(m1), digits = 2)
```

---

orbbound *Sensitivity Analysis for Outcome Reporting Bias (ORB)*

Description

Implementation of the method by Copas & Jackson (2004) to evaluate outcome reporting bias in meta-analysis. An upper bound for outcome reporting bias is estimated for a given number of studies suspected with outcome reporting bias.

Usage

```
orbbound(x, k.suspect = 1, tau = x$tau, left = NULL, backtransf = x$backtransf)
```

Arguments

`x`	An object of class `meta`.
`k.suspect`	Number of studies with suspected outcome reporting bias.
`tau`	Square-root of between-study variance tau-squared.
`left`	A logical indicating whether the cause of any selection bias is due to missing studies on the left or right of the funnel plot: left hand side if `left=TRUE`, right hand side if `left=FALSE`. If not set, the linear regression test for funnel plot asymmetry (i.e., function `metabias(...,meth="linreg")`) is used to determine whether studies are missing on the left or right hand side.

backtransf — A logical indicating whether results should be back transformed in printouts and plots. If backtransf=TRUE (default), results for sm="OR" are printed as odds ratios rather than log odds ratio, for example.

Details

This function provides the method by Copas and Jackson (2004) to estimate an upper bound for bias for a given number of studies with suspected outcome reporting bias.

Based on the upper bound of outcome reporting bias, treatment estimates and confidence limits adjusted for bias are calculated.

For comparison, the original meta-analysis is always considered in the sensitivity analysis (i.e. value 0 is always added to k.suspect).

Value

An object of class c("orbbound") with corresponding print and forest function. The object is a list containing the following components:

k.suspect, tau	As defined above.
maxbias	Maximum bias for given values of k.suspect.
fixed	Adjusted treatment estimates and corresponding quantities for fixed effect model (a list with elements TE, seTE, lower, upper, z, p, level, df).
random	Adjusted treatment estimates and corresponding quantities for random effects model (a list with elements TE, seTE, lower, upper, z, p, level, df).
left	Whether selection bias expected on left or right
x	Meta-analysis object (i.e. argument x from function call).
call	Function call.
version	Version of R package metasens used to create object.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

References

Copas J, Jackson D (2004): A bound for publication bias based on the fraction of unpublished studies. *Biometrics*, **60**, 146–53

See Also

forest.orbbound, print.orbbound

Examples

```
data(Fleiss1993bin, package = "meta")

m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")

orb1 <- orbbound(m1, k.suspect = 1:5)
print(orb1, digits = 2)
forest(orb1, xlim = c(0.75, 1.5))

# Same result
#
orb2 <- orbbound(m1, k.suspect = 1:5, left = FALSE)
print(orb2, digits = 2)

# Assuming bias in other direction
#
orb3 <- orbbound(m1, k.suspect = 1:5, left = TRUE)
print(orb3, digits = 2)
```

---

plot.copas *Display results of Copas selection modelling*

Description

Four plots (selectable by 'which') are currently available: (1) funnel plot, (2) contour plot, (3) treatment effect plot, (4) p-value for residual publication bias plot. By default, all plots are provided.

Usage

```
## S3 method for class 'copas'
plot(
  x,
  which = 1:4,
  caption = c("Funnel plot", "Contour plot", "Treatment effect plot",
    "P-value for residual selection bias"),
  xlim.pp = NULL,
  orthogonal.line = TRUE,
  lines = FALSE,
  warn = -1,
  ...
)
```

Arguments

x	An object of class copas, generated by the copas function
which	Specify plots required: 1:4 produces all plots (default); 3 produces plot 3 etc; c(1,3) produces plots 1 and 3, and so on.

`caption` Specify plot captions. Note that four captions must be specified even if fewer graphs are displayed (which is the case if the predefined captions are utilised). This must be considered if user-defined captions are provided. Captions corresponding to plots that are not displayed can be left empty. For example, if only plot 3 is selected, we might specify `caption=c("","","Plot 3","")`.

`xlim.pp` A vector of x-axis limits for plots 3 and 4, i.e. for the probability of publishing the study with largest standard deviation. E.g. to specify limits between 0.3 and 0.1 set `xlim.pp=c(0.3,0.1)`.

`orthogonal.line` A logical indicating whether the orthogonal line should be displayed in plot 2 (contour plot).

`lines` (Diagnostic use only) A logical indicating whether regression lines should be plotted in contour plot. These regression lines attempt to summarise each contour of constant treatment effect by a straight line, prior to calculating the orthogonal line. Regression lines with a positive adjusted `R^2` will be printed in green color, others will be printed in red color.

`warn` A number setting the handling of warning messages. It is not uncommon for numerical problems to be encountered during estimation over the grid of (gamma0, gamma1) values. Usually this does not indicate a serious problem. This option specifies what to do with warning messages. `warn=-1`: ignore all warnings; `warn=0` (the default): store warnings till function finishes; if there are less than 10, print them, otherwise print a message saying warning messages were generated; `warn=1`: print warnings as they occur; `warn=2`: stop the function when the first warning is generated. For further details see `help(options)`.

`...` other arguments to the function will be ignored (this option only included to conform with R standards)

Details

Takes an object created by the `copas` function and draws up to four plots to display the results of the Copas selection modelling.

The argument `which` specifies the plots to be drawn; plot numbers below will be produced by setting `which=1`, etc.

Plot 1: Funnel plot of studies in meta-analysis. Vertical grey line is usual random effects estimate (DerSimonian-Laird method); vertical broken line is fixed effects estimate.

Plot 2: Plot of contours of treatment effect (estimated by the Copas model) as the selection probability varies (the selection probability is a function of gamma0 and gamma1 - see `help(copas)` or the reference below).

Plot 3: Assuming the contours of treatment effect in Plot 2 are locally parallel, the results can be summarised in terms of the probability of publishing the study with the largest standard error. This plot displays the results of doing this, showing how the estimated treatment effect (and `100*level`% confidence interval) vary as the probability of publishing the study with the largest standard error decreases.

The three horizontal grey lines are the usual random effects treatment estimate (center) +/- the `100*level`% confidence interval (upper/lower grey lines).

Plot 4: For any degree of selection (i.e. probability of the study with largest SE being published), we can calculate a p-value for the hypothesis that no further selection remains unexplained in the data. These plot displays these p-values against the probability that the study with the largest SE is published.

Under the copas selection model, probabilities of the smallest study being published which correspond to p-values for residual selection bias that are larger than 0.1 are more plausible. The corresponding treatment effect in plot 3 is thus the most plausible under the copas selection model.

Note

In the current version, fine control of the graphics parameters for the individual panels is not possible. However, all the data used to create the plots can be extracted manually from the object created by the copas function (see attributes list for copas) and used to create tailor-made plots.

Author(s)

James Carpenter <James.Carpenter@lshtm.ac.uk>, Guido Schwarzer <sc@imbi.uni-freiburg.de>

References

Carpenter JR, Schwarzer G, Rücker G, Künstler R (2009): Empirical evaluation showed that the Copas selection model provided a useful summary in 80% of meta-analyses. *Journal of Clinical Epidemiology*, **62**, 624–31

Schwarzer G, Carpenter J, Rücker G (2010): Empirical evaluation suggests Copas selection model preferable to trim-and-fill method for selection bias in meta-analysis. *Journal of Clinical Epidemiology*, **63**, 282–8

See Also

copas, summary.copas, metabias, metagen

Examples

```
data(Fleiss1993bin, package = "meta")

# Perform meta-analysis (outcome measure is OR = odds ratio)
#
m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")

# Perform Copas analysis
#
cop1 <- copas(m1)

# Plot results
#
plot(cop1)

# Only show plots 1 and 2 (without orthogonal line)
#
plot(cop1, which = 1:2, orth = FALSE)

# Another example showing use of more arguments
```

```
# Note the use of "\n" to create a new line in the caption
#
plot(cop1,
     which = 3,
     caption = c("", "",
                 "Variation in estimated treatment\n effect with selection",
                 ""),
     xlim.pp = c(1, 0.5))
```

`print.copas` *Print method for Copas selection model*

Description

Print method for objects of class `copas`.

Usage

```
## S3 method for class 'copas'
print(
  x,
  backtransf = x$backtransf,
  digits = gs("digits"),
  digits.se = gs("digits.se"),
  ...
)
```

Arguments

`x`	An object of class `copas`.
`backtransf`	A logical indicating whether results should be back transformed in printouts and plots. If `backtransf=TRUE` (default), results for `sm="OR"` are printed as odds ratios rather than log odds ratio, for example.
`digits`	Minimal number of significant digits, see `print.default`.
`digits.se`	Minimal number of significant digits for standard deviations and standard errors, see `print.default`.
`...`	other arguments to the function will be ignored (this option included only to conform with R standards)

Details

This function prints the following information:

Range of gamma0 values used (see `help(copas)`);

Range of gamma1 values used (see `help(copas)`);

Largest SE of all studies in meta-analysis;

Range of probability publishing trial with largest SE;

The next table gives details relating to the summary of the contour plot. Specifically, it gives details from fitting a straight line to each treatment-contour in the contour plot. Column 1 (headed level) shows the treatment-contours; column 2 (nobs) shows the number of observations used by the contour plot command within the copas function to plot this contour line; column 3 (adj.r.square) shows the adjusted r-square from fitting a straight line to this contour; columns 4 & 5 show the slope and its standard error from fitting a straight line to this contour.

Next, the printout of summary.copas is shown.

Author(s)

James Carpenter <James.Carpenter@lshtm.ac.uk>, Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

copas, plot.copas, summary.copas

Examples

```
data(Fleiss1993bin, package = "meta")

# Perform meta analysis, effect measure is odds ratio (OR)
#
m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")

# Perform Copas analysis
#
cop1 <- copas(m1)
cop1
```

---

print.limitmeta *Print method for limit meta-analysis*

Description

Print method for objects of class limitmeta.

Usage

```
## S3 method for class 'limitmeta'
print(
  x,
  sortvar,
  backtransf = x$backtransf,
  digits = gs("digits"),
  big.mark = gs("big.mark"),
  ...
)
```

Arguments

x	An object of class limitmeta
sortvar	An optional vector used to sort the individual studies (must be of same length as x$TE).
backtransf	A logical indicating whether results should be back transformed in printouts and plots. If backtransf=TRUE (default), results for sm="OR" are printed as odds ratios rather than log odds ratio, for example.
digits	Minimal number of significant digits, see print.default.
big.mark	A character used as thousands separator.
...	Additional arguments which are passed on to print.summary.limitmeta called internally.

Details

This function prints the summary information from summary.limitmeta together with the following study information:

- Effect estimate with confidence interval
- Shrunken effect estimates with confidence interval

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

limitmeta, summary.limitmeta

Examples

```
data(Moore1998)
m1 <- metabin(succ.e, nobs.e, succ.c, nobs.c,
              data = Moore1998, sm = "OR", method = "Inverse")

print(limitmeta(m1), digits = 2)
```

---

print.orbbound *Print method for objects of class orbbound*

Description

Print method for objects of class orbbound.

Usage

```
## S3 method for class 'orbbound'
print(
  x,
  comb.fixed = x$x$comb.fixed,
  comb.random = x$x$comb.random,
  header = TRUE,
  backtransf = x$backtransf,
  digits = gs("digits"),
  digits.stat = gs("digits.stat"),
  digits.pval = max(gs("digits.pval"), 2),
  digits.tau2 = gs("digits.tau2"),
  scientific.pval = gs("scientific.pval"),
  big.mark = gs("big.mark"),
  ...
)
```

Arguments

x	An object of class orbbound.
comb.fixed	A logical indicating whether sensitivity analysis for fixed effect model should be printed.
comb.random	A logical indicating whether sensitivity analysis for random effects model should be printed.
header	A logical indicating whether information on meta-analysis should be printed at top of printout.
backtransf	A logical indicating whether printed results should be back transformed. If backtransf=TRUE, results for sm="OR" are printed as odds ratios rather than log odds ratios and results for sm="ZCOR" are printed as correlations rather than Fisher's z transformed correlations, for example.
digits	Minimal number of significant digits, see print.default.
digits.stat	Minimal number of significant digits for z- or t-value, see print.default.
digits.pval	Minimal number of significant digits for p-value of overall treatment effect, see print.default.
digits.tau2	Minimal number of significant digits for between-study variance, see print.default.
scientific.pval	A logical specifying whether p-values should be printed in scientific notation, e.g., 1.2345e-01 instead of 0.12345.
big.mark	A character used as thousands separator.
...	Additional arguments

Details

For summary measures 'RR', 'OR', and 'HR' column labeled maxbias contains the relative bias, e.g. a value of 1.10 means a maximum overestimation by 10 percent. If logscale=TRUE for these summary measures, maximum bias is instead printed as absolute bias.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

orbbound, forest.orbbound

Examples

```
data(Fleiss1993bin, package = "meta")

m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")

orb1 <- orbbound(m1, k.suspect = 1:5)
print(orb1, digits = 2)

# Print log odds ratios instead of odds ratios
#
print(orb1, digits = 2, backtransf = FALSE)

# Assuming that studies are missing on the left side
#
orb1.missleft <- orbbound(m1, k.suspect = 1:5, left = TRUE)
orb1.missleft

m2 <- metabin(d.asp, n.asp, d.plac, n.plac,
              data = Fleiss1993bin, sm = "OR", method = "Inverse")

orb2 <- orbbound(m2, k.suspect = 1:5)
print(orb2, digits = 2)
```

---

print.summary.copas *Print method for summary of Copas selection model*

Description

Print method for objects of class summary.copas.

Usage

```
## S3 method for class 'summary.copas'
print(
  x,
  backtransf = x$backtransf,
  digits = gs("digits"),
  digits.pval = max(gs("digits.pval"), 2),
  digits.prop = gs("digits.prop"),
  scientific.pval = gs("scientific.pval"),
```

```
  big.mark = gs("big.mark"),
  header = TRUE,
  ...
)
```

Arguments

x	An object of class summary.copas.
backtransf	A logical indicating whether results should be back transformed in printouts and plots. If backtransf=TRUE (default), results for sm="OR" are printed as odds ratios rather than log odds ratio, for example.
digits	Minimal number of significant digits, see print.default.
digits.pval	Minimal number of significant digits for p-value of overall treatment effect, see print.default.
digits.prop	Minimal number of significant digits for proportions, see print.default.
scientific.pval	A logical specifying whether p-values should be printed in scientific notation, e.g., 1.2345e-01 instead of 0.12345.
big.mark	A character used as thousands separator.
header	A logical indicating whether information on title of meta-analysis, comparison and outcome should be printed at the beginning of the printout.
...	other arguments to the function will be ignored (this option included only to conform with R standards)

Details

This function prints a summary of a Copas analysis, performed using the function copas. It complements the graphical summary of the results, generated using plot.copas.

Specifically it prints a table where the:

first column corresponds to the x-axis in plots 3 & 4 from plot.copas;

second column corresponds to the treatment effect displayed in plot 3 from plot.copas;

third and fourth columns give the confidence intervals for this treatment effect,

fifth colum gives the p-value for an overall treatment effect,

sixth column gives the p-value for residual publication bias (the y-axis of plot 4 from plot.copas (see help(plot.copas) under plot 4 for a further explanation of this p-value))

seventh column gives an approximate estimate of the number of studies the model suggests remain unpublished if the probability of publishing the study with the largest SE is as in column 1.

Below this is displayed the results of the Copas analysis for the smallest degree of selection for which the p-value for evidence of residual selection bias exceeds sign.rsb (default: 0.1). This is simply extracted from the corresponding row in the table above.

Lastly, the usual random effects estimate (based on the DerSimonian-Laird method) and 95% confidence interval is printed.

Author(s)

James Carpenter <James.Carpenter@lshtm.ac.uk>, Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

copas, plot.copas, summary.copas

Examples

```
data(Fleiss1993bin, package = "meta")

# Perform meta analysis, effect measure is odds ratio (OR)
#
m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data=Fleiss1993bin, sm="OR")

# Print summary of Copas analysis
#
summary(copas(m1), level=0.95)
```

---

print.summary.limitmeta

Print method for summary of limit meta-analysis

Description

Print method for objects of class summary.limitmeta.

Usage

```
## S3 method for class 'summary.limitmeta'
print(
  x,
  backtransf = x$backtransf,
  digits = gs("digits"),
  header = TRUE,
  pscale = x$x$pscale,
  irscale = x$x$irscale,
  irunit = x$x$irunit,
  digits.stat = gs("digits.stat"),
  digits.pval = gs("digits.pval"),
  digits.Q = gs("digits.Q"),
  digits.tau2 = gs("digits.tau2"),
  digits.I2 = gs("digits.I2"),
  scientific.pval = gs("scientific.pval"),
  big.mark = gs("big.mark"),
  print.Rb = gs("print.Rb"),
  warn.backtransf = FALSE,
```

```
  ...
)
```

Arguments

x	An object of class summary.limitmeta.
backtransf	A logical indicating whether results should be back transformed in printouts and plots. If backtransf=TRUE (default), results for sm="OR" are printed as odds ratios rather than log odds ratio, for example.
digits	Minimal number of significant digits, see print.default.
header	A logical indicating whether information on title of meta-analysis, comparison and outcome should be printed at the beginning of the printout.
pscale	A numeric giving scaling factor for printing of single event probabilities, i.e. if argument sm is equal to "PLOGIT", "PLN", "PRAW", "PAS", or "PFT".
irscale	A numeric defining a scaling factor for printing of rates, i.e. if argument sm is equal to "IR", "IRLN", "IRS", or "IRFT".
irunit	A character specifying the time unit used to calculate rates, e.g. person-years.
digits.stat	Minimal number of significant digits for z- or t-value, see print.default.
digits.pval	Minimal number of significant digits for p-value of overall treatment effect, see print.default.
digits.Q	Minimal number of significant digits for heterogeneity statistic Q, see print.default.
digits.tau2	Minimal number of significant digits for between-study variance, see print.default.
digits.I2	Minimal number of significant digits for I-squared and Rb statistic, see print.default.
scientific.pval	A logical specifying whether p-values should be printed in scientific notation, e.g., 1.2345e-01 instead of 0.12345.
big.mark	A character used as thousands separator.
print.Rb	A logical specifying whether heterogeneity statistic Rb should be printed.
warn.backtransf	A logical indicating whether a warning should be printed if backtransformed proportions and rates are below 0 and backtransformed proportions are above 1.
...	Additional arguments (ignored).

Details

This function prints summary information for a limit meta-analysis (Rücker et al., 2011); unadjusted as well as adjusted effect estimates in a random effects model are printed.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

limitmeta, summary.limitmeta

Examples

```
data(Moore1998)
m1 <- metabin(succ.e, nobs.e, succ.c, nobs.c,
              data = Moore1998, sm = "OR", method = "Inverse")

print(summary(limitmeta(m1)), digits = 2)
```

`summary.copas` *Summary method for Copas selection model*

Description

Summary method for objects of class `copas`.

Usage

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

Arguments

`object`	An object of class `copas`.
`...`	other arguments to the function will be ignored (this option included only to conform with R standards)

Details

This function complements the graphical summary of the results of a Copas selection model, generated using `plot.copas`.

Value

An object of class "summary.copas" with corresponding print function. The object is a list containing the following components:

`slope`	Results for points on orthogonal line (a list with elements TE, seTE, lower, upper, statistic, p, level).
`publprob`	Vector of probabilities of publishing the smallest study.
`pval.rsb`	P-values for tests on presence of residual selection bias
`N.unpubl`	Approximate number of studies the model suggests remain unpublished
`adjust`	Result of Copas selection model adjusted for selection bias (a list with elements TE, seTE, lower, upper, statistic, p, level).
`sign.rsb`	The significance level for the test of residual selection bias.
`pval.rsb.adj`	P-value for test on presence of residual selection bias for adjusted effect given in `adjust`.

N.unpubl.adj	Approximate number of studies the model suggests remain unpublished for adjusted effect given in adjust
random	Results for usual random effects model (a list with elements TE, seTE, lower, upper, statistic, p, level).
sm	A character string indicating underlying summary measure.
ci.lab	Label for confidence interval.
title	Title of meta-analysis / systematic review.
complab	Comparison label.
outclab	Outcome label.
version	Version of R package metasens used to create object.

Author(s)

James Carpenter <James.Carpenter@lshtm.ac.uk>, Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

copas, plot.copas, metabias, metagen

Examples

```
data(Fleiss1993bin, package = "meta")

# Perform meta analysis, effect measure is odds ratio (OR)
#
m1 <- metabin(d.asp, n.asp, d.plac, n.plac, data = Fleiss1993bin, sm = "OR")

# Print summary of Copas analysis
#
summary(copas(m1, level.comb = 0.95))
```

---

summary.limitmeta *Summary method for limit meta-analysis*

Description

Summary method for objects of class limitmeta.

Usage

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

Arguments

object	An object of class limitmeta.
...	Additional arguments (ignored).

Value

This function returns the same list as the function limitmeta, however class "summary.limitmeta" is added to the object in order to print a short summary of the limit meta-analysis object.

Author(s)

Guido Schwarzer <sc@imbi.uni-freiburg.de>

See Also

limitmeta, funnel.limitmeta, print.summary.limitmeta

Examples

```
data(Moore1998)
m1 <- metabin(succ.e, nobs.e, succ.c, nobs.c,
              data = Moore1998, sm = "OR", method = "Inverse")

summary(limitmeta(m1))
```

Index