Package ‘nmathresh’

March 19, 2020

Type Package

Title Thresholds and Invariant Intervals for Network Meta-Analysis

Version 0.1.5

Description Calculation and presentation of decision-invariant bias adjustment thresholds and intervals for Network Meta-Analysis, as described by Phillippo et al. (2018) <doi:10.1111/rssa.12341>. These describe the smallest changes to the data that would result in a change of decision.

License GPL-3

Encoding UTF-8

LazyData TRUE

Imports nnls, grid, gridExtra, gtable, Matrix, ggplot2

Depends R (>= 3.1.0)

RoxygenNote 7.0.2

Suggests knitr, rmarkdown, coda

VignetteBuilder knitr

NeedsCompilation no

Author David Phillippo [aut, cre]

Maintainer David Phillippo <david.phillippo@bristol.ac.uk>

Repository CRAN

Date/Publication 2020-03-19 15:40:05 UTC

*R topics documented:

d_ab2i ................................................................. 2
get.int ................................................................. 3
nma_thresh ........................................................... 4
recon_vcov ........................................................... 6
SocAnx.post.cov .................................................... 8
SocAnx.post.summary ............................................. 9
thresh-class ......................................................... 10
d_ab2i

Convert contrast indexing

Description
Functions for converting between \( d_{ab} \) indexing of contrasts (useful notationally) and \( d[i] \) indexing used by R.

Usage
\[
d_{ab2i}(a, b, K) \\
d_{i2ab}(i, K)
\]

Arguments
- \( a \): Vector of treatment codes \( a \).
- \( b \): Vector of treatment codes \( b \).
- \( K \): Total number of treatments.
- \( i \): Vector of indices \( i \).

Value
\( d_{ab2i} \) returns a vector of indices \( i \). \( d_{i2ab} \) returns a data frame of indices \( a \) and \( b \).

Functions
- \( d_{ab2i} \): Convert \( d[i] \) type indices to \( d_{ab} \) type indices.
- \( d_{i2ab} \): Convert \( d_{ab} \) type indices to \( d[i] \) type indices.

Note
By convention, \( 1 \leq a < b \leq K \). If this is not the case, an error will be thrown. For a given number of treatments \( K \), the total number of possible contrasts \( d_{ab} \) is \( K(K - 1)/2 \), and hence \( i \leq K \). Again, if this is not the case, an error will be thrown.

Examples
\[
d_{ab2i}(c(1,1,1, 2,2, 3), c(2,3,4, 3,4, 4), K=4) \\
d_{i2ab}(1:6, K=4)
\]
get.int

Get thresholds from U matrix

Description

Return the positive and negative thresholds for an observation, given a vector of possible threshold solutions. This function is intended for internal use, and is called by nma_thres automatically.

Usage

get.int(
  x,
  kstar,
  trt.code,
  contrs,
  mcid = FALSE,
  mean.dk = NULL,
  inflmat = NULL,
  opt.max = NULL
)

Arguments

x Column of U matrix, containing all possible threshold solutions for a data point.
kstar Base-case optimal treatment.
trt.code Vector of (possibly recoded) treatments. See nma_thres parameter of the same name.
contrs Details of contrasts corresponding to rows in x, as rows of the data.frame output by d_i2ab.
mcid Use MCID decision rule? Default FALSE.
mean.dk Posterior means of basic treatment parameters, required when mcid is TRUE.
inflmat Column of influence matrix H for the data point, required when mcid is TRUE.
opt.max Is the maximum treatment effect optimal? See nma_thres parameter of same name. Required when mcid is TRUE.

Value

Data frame of thresholds and new optimal treatments with columns lo, lo.newkstar, hi, and hi.newkstar.
**nma_thres**  
*Calculate thresholds and invariant intervals*

**Description**

This function calculates decision-invariant bias-adjustment thresholds and intervals for Bayesian network meta-analysis, as described by Phillippo *et al.* (2018). Thresholds are derived from the joint posterior, and reflect the amount of change to a data point before the treatment decision changes. Calculation is achieved using fast matrix operations.

**Usage**

```r
nma_thres(
  mean.dk,  
  lhood,    
  post,     
  nmatype = "fixed",  
  X = NULL,  
  mu.design = NULL,  
  delta.design = NULL,  
  opt.max = TRUE,  
  trt.rank = 1,  
  trt.code = NULL,  
  trt.sub = NULL,  
  mcid = 0,  
  mcid.type = "decision"
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean.dk</td>
<td>Posterior means of basic treatment parameters $d_k$.</td>
</tr>
<tr>
<td>lhood</td>
<td>Likelihood (data) covariance matrix.</td>
</tr>
<tr>
<td>post</td>
<td>Posterior covariance matrix (see details).</td>
</tr>
<tr>
<td>nmatype</td>
<td>Character string, giving the type of NMA performed. One of &quot;fixed&quot; (fixed effects, the default) or &quot;random&quot; (random effects). May be abbreviated.</td>
</tr>
<tr>
<td>X</td>
<td>[FE models only] Design matrix for basic treatment parameters.</td>
</tr>
<tr>
<td>mu.design</td>
<td>[RE models only] Design matrix for any extra parameters. Defaults to NULL (no extra parameters).</td>
</tr>
<tr>
<td>delta.design</td>
<td>[RE models only] Design matrix for delta, defaults to the $N \times N$ identity matrix.</td>
</tr>
<tr>
<td>opt.max</td>
<td>Should the optimal decision be the maximal treatment effect (TRUE, default) or the minimum (FALSE).</td>
</tr>
<tr>
<td>trt.rank</td>
<td>Rank of the treatment to derive thresholds for. Defaults to 1, thresholds for the optimum treatment.</td>
</tr>
<tr>
<td>trt.code</td>
<td>Treatment codings of the reference treatment and in the parameter vector $d_k$. Use if treatments re-labelled or re-ordered. Default is equivalent to 1:K.</td>
</tr>
</tbody>
</table>
nma_thresh

trt.sub

Only look at thresholds in this subset of treatments in trt.code, e.g. if some are excluded from the ranking. Default is equivalent to 1:K.

mcid

Minimal clinically important difference for the decision (when mcid.type = 'decision') or for changing the decision (when mcid.type = 'change'). Defaults to 0, use the maximal efficacy decision rule.

mcid.type

Default 'decision', the decision rule is based on MCID (see details). Otherwise 'change', use the maximum efficacy rule, but only consider changing the decision when the alternative treatment becomes more effective than the base case by mcid or more.

Details

This function provides bias-adjustment threshold analysis for both fixed and random effects NMA models, as described by Phillippo et al. (2018). Parameters mean.dk, lhood, and post are always required, however there are differences in the specification of post and other required parameters and between the fixed and random effects cases:

**FE models**  The design matrix $X$ for basic treatment parameters is required. The posterior covariance matrix specified in post should only refer to the basic treatment parameters.

**RE models**  The design matrix $mu$.design for additional parameters (e.g. covariates) is required, as is the design matrix delta.design for random effects terms. The posterior covariance matrix specified in post should refer to the basic treatment parameters, RE terms, and additional parameters in that order; i.e. post should be the posterior covariance matrix of the vector $(d^T, \delta^T, \mu^T)^T$.

Value

An object of class thresh.

Model details

**The FE NMA model**

The fixed effects NMA model is assumed to be of the form

Prior:  $d \sim N(d_0, \Sigma_d)$

Likelihood:  $y|d \sim N(\delta, V)$

FE model:  $\delta = Xd + M\mu$

The additional parameters $\mu$ may be given any sensible prior; they do not affect the threshold analysis in any way.

**The RE NMA model**

The random effects NMA model is assumed to be of the form

Priors:  $d \sim N(d_0, \Sigma_d), \mu \sim N(\mu_0, \Sigma_\mu)$

Likelihood:  $y|d, \mu, \tau^2 \sim N(L\delta + M\mu, V)$

RE model:  $\delta \sim N(Xd, \tau^2)$

The between-study heterogeneity parameter $\tau^2$ may be given any sensible prior. In the RE case, the threshold derivations make the approximation that $\tau^2$ is fixed and known.
Decision rules

The default decision rule is maximal efficacy; the optimal treatment is \( k^* = \arg\max_k E(d_k) \).

When \( \epsilon = \text{mcid} \) is greater than zero and \( \text{mcid.type} = '\text{decision}' \), the decision rule is no longer for a single best treatment, but is based on minimal clinically important difference. A treatment is in the optimal set if \( E(d_k) \geq \epsilon \) and \( \max_a E(d_a) - E(d_k) \leq \epsilon \).

When \( \text{mcid.type} = '\text{change}' \), the maximal efficacy rule is used, but thresholds are found for when a new treatment is better than the base-case optimal by at least \( \text{mcid} \).

See Also

recon_vcov, thresh_forest, thresh-class.

Examples

# Please see the vignette "Examples" for worked examples including use of
# this function, including more information on the brief code below.

vignette("Examples", package = "nmathresh")

### Contrast level thresholds for Thrombolytic treatments NMA
K <- 6  # Number of treatments

# Contrast design matrix is
X <- matrix(ncol = K-1, byrow = TRUE,
   c(1, 0, 0, 0, 0,
     0, 1, 0, 0, 0,
     0, 0, 1, 0, 0,
     0, 0, 0, 1, 0,
     0, -1, 1, 0, 0,
     0, -1, 0, 1, 0,
     0, -1, 0, 0, 1))

# Reconstruct hypothetical likelihood covariance matrix using NNLS
lik.cov <- recon_vcov(Thrombo.post.cov, prior.prec = .0001, X = X)

# Thresholds are then
thresh <- nma_thresh(mean.dk = Thrombo.post.summary$statistics[1:(K-1), "Mean"],
   lhood = lik.cov,
   post = Thrombo.post.cov,
   nmatype = "fixed",
   X = X,
   opt.max = FALSE)
Description

Reconstruct the contrast-level likelihood covariance matrix from prior and posterior covariance matrices. The resulting likelihood covariance matrix can then be used to perform a contrast-level threshold analysis with the function `nma_thresh`.

Usage

```r
recon_vcov(
  post,
  prior.prec = 1e-04,
  prior.vcov = diag(1/prior.prec, dim(post)[1]),
  X = NULL,
  verbose = FALSE
)
```

Arguments

- **post**: Posterior covariance matrix.
- **prior.prec**: Prior precision. Defaults to .0001 which is a common flat prior for NMA. Not used if `prior.vcov` is specified.
- **prior.vcov**: Prior covariance matrix. Defaults to a diagonal matrix of the same size as `post`, with elements \(1/prior.prec\).
- **X**: Contrast design matrix. If omitted a complete network is assumed.
- **verbose**: Print intermediate matrices? Defaults to `FALSE`.

Details

Full details of the calculation are given by Phillippo et al. (2018). Briefly, the aim is to recover the contrast-level likelihood covariance matrix \(V\) that would have led to the posterior covariance matrix \(\Sigma\) being obtained from a fixed effects NMA, with design matrix \(X\) and prior covariance matrix \(\Sigma_d\) for a normal prior on the basic treatment parameters. This is possible in this case via the equation (resulting from conjugacy):

\[
\Sigma^{-1} = X^T V^{-1} X + \Sigma_d^{-1}.
\]

When the treatment network is complete (i.e. fully connected), this equation may be rearranged exactly.

When the treatment network is incomplete (i.e. not all treatments are directly compared), this equation may be solved through the use of non-negative least squares (NNLS).

When NNLS is used, some additional diagnostics are printed (and returned as attributes). Firstly, the residual sum-of-squares (RSS) from the NNLS fit. The RSS is further split into fixed RSS, from structural zeros in the reconstructed posterior according to the design matrix (and hence not fitted) that are non-zero in the true posterior, and fitted RSS, from the other fitted elements. Secondly, the Kullback-Leibler divergence of the reconstructed posterior from the true posterior. Interpreting the KL divergence as a log Bayes factor, values less than 1 indicate negligible differences between the reconstructed posterior from the true posterior, whilst values greater than 3 indicate considerable differences.
Value
A matrix; the reconstructed likelihood covariance matrix. If NNLS is used, the residual sum-of-squares and Kullback-Leibler divergence diagnostics (as printed to the console) are returned as additional attributes \( \text{rss.total, rss.fixed, rss.free, kl.divergence} \).

See Also
nma_thresh.

Examples
# Please see the vignette "Examples" for worked examples including use of # this function, including more information on the brief code below.

vignette("Examples", package = "nmathresh")

### Contrast level thresholds for Thrombolytic treatments NMA

K <- 6 # Number of treatments

# Contrast design matrix is
X <- matrix(ncol = K-1, byrow = TRUE,
            c(1, 0, 0, 0, 0,
              0, 1, 0, 0, 0,
              0, 0, 1, 0, 0,
              0, 0, 0, 1, 0,
              0, -1, 1, 0, 0,
              0, -1, 0, 1, 0,
              0, -1, 0, 0, 1))

# Reconstruct hypothetical likelihood covariance matrix using NNLS
lik.cov <- recon_vcov(Thrombo.post.cov, prior.prec = .0001, X = X)
SocAnx.post.summary

Source
Generated from WinBUGS output, using the WinBUGS code from Mayo-Wilson et al. (2014). See also vignette("Examples",package = "nmathresh").

References

See Also
SocAnx.post.summary

SocAnx.post.summary  Posterior summary from Social Anxiety NMA

Description
A summary.mcmc object of the type produced by the coda package, containing the requisite posterior summary information on the variables d (basic treatment effect parameters), delta (shrunken random effects estimates for each study), and diff (contrasts of treatment effect parameters).

Usage
SocAnx.post.summary

Format
A summary.mcmc object. The key components for our use are:

statistics Matrix containing the posterior summary statistics of the variables d, delta, and diff, with columns for Mean, SD, Naive SE, and Time-series SE (also known as the Monte-Carlo standard error)
quantiles Matrix containing the posterior 2.5%, 25%, 50%, 75%, and 97.5% quantiles of the variables d, delta, and diff

Source
Generated from WinBUGS output, using the WinBUGS code from Mayo-Wilson et al. (2014). See also vignette("Examples",package = "nmathresh").

References
See Also

* summary.mcmc, SocAnx.post.cov

---

**thresh-class**

**The thresh class**

**Description**

The function `nmathresh` returns S3 objects of class `thresh`.

**Details**

Objects of class `thresh` have the following components:

- **thresholds**  A data frame with columns `lo` and `hi` for the lower and upper thresholds, and `lo.newkstar` and `hi.newkstar` for the new optimal (or rank-trt.rank) treatments at each of the thresholds.

- **U**  The threshold solutions matrix. One column for each data point `m`, one row for each contrast `d_{ab}` (in ascending order). The elements `U_{ab,m}` describe the amount of adjustment to data point `y_m` required to reverse the relative ranking of treatments `a` and `b`. This matrix is particularly useful for deriving thresholds for more complex decisions (e.g. bias-adjustment thresholds for a new treatment entering the top two, for any change in rank of the top three, etc.).

- **Ukstar**  The threshold solutions matrix limited to contrasts involving `k*`. In other words, the rows of `U` corresponding to contrasts of the form `d_{ak*}` or `d_{k*a}`. Elements `U_{ak*,m}` of this matrix describe the amount of adjustment to data point `y_m` required to make treatment `a` optimal (or rank-trt.rank) over `k*`.

- **H**  The influence matrix of the data on the basic treatment parameters. One column for each data point `m`, one row for each basic treatment parameter `d_k`. Elements `H_{k,m}` describe the influence of data point `y_m` on parameter `d_k`. This matrix can be used to derive more complex thresholds (e.g. 2D thresholds for simultaneous adjustments to two data points, or thresholds for common adjustments to a group of data points).

- **kstar**  The base-case optimal (or rank-trt.rank) treatment `k*`.

- **call**  A list containing all the arguments defined in the original call to `nma_thresh`.

**See Also**

* nma_thresh
thresh_2d  Producing two-dimensional invariant regions

Description
This function produces two-dimensional threshold lines and invariant regions, as shown by Phillippo et al. (2018).

Usage
thresh_2d(
  thresh,
  idx,
  idy,
  xlab = paste("Adjustment to data point", idx),
  ylab = paste("Adjustment to data point", idy),
  xlim = NULL,
  ylim = NULL,
  breaks = waiver(),
  xbreaks = breaks,
  ybreaks = breaks,
  fill = rgb(0.72, 0.8, 0.93, 0.7),
  lwd = 1,
  fontsize = 12
)

Arguments
thresh A thresh object, as produced by nma_thresh.
idx Integer specifying the index (with respect to thresh$thresholds) of the first data point to consider adjusting. Will be shown on the x axis.
idy Integer specifying the index (with respect to thresh$thresholds) of the second data point to consider adjusting. Will be shown on the y axis.
xlab Character string giving the label for the x axis.
ylab Character string giving the label for the y axis.
xlim Numeric vector of length 2, giving the x axis limits.
ylim Numeric vector of length 2, giving the y axis limits.
breaks Numeric vector giving position of tick marks on the x and y axes. Calculated automatically by default.
xbreaks Numeric vector giving position of tick marks on the x axis. Equal to breaks by default, if set this overrides any value given to breaks.
ybreaks Numeric vector giving position of tick marks on the y axis. Equal to breaks by default, if set this overrides any value given to breaks.
fill Fill colour for invariant region. Defaults to a nice shade of blue rgb(0.72, 0.8, 0.93, 0.7).
lwd Line width for threshold lines. Default 1.
fontsize Font size for labels. Default 12.
Value

A ggplot object containing the 2D threshold plot, which is returned invisibly and plotted (unless assigned).

Examples

# Please see the vignette "Examples" for worked examples including use of
# this function, including more information on the brief code below.

vignette("Examples", package = "nmathresh")

### Contrast level thresholds for Thrombolytic treatments NMA

K <- 6  # Number of treatments

# Contrast design matrix is

X <- matrix(ncol = K-1, byrow = TRUE,
            c(1, 0, 0, 0, 0,
             0, 1, 0, 0, 0,
             0, 0, 1, 0, 0,
             0, 0, 0, 1, 0,
             0, -1, 1, 0, 0,
             0, -1, 0, 1, 0,
             0, -1, 0, 0, 1))

# Reconstruct hypothetical likelihood covariance matrix using NNLS

lik.cov <- recon_vcov(Thrombo.post.cov, prior.prec = .0001, X = X)

# Thresholds are then

thresh <- nma_thresh(mean.dk = Thrombo.post.summary$statistics[1:(K-1), "Mean"],
                    lhood = lik.cov,
                    post = Thrombo.post.cov,
                    nmatype = "fixed",
                    X = X,
                    opt.max = FALSE)

# Produce an invariant region for simultaneous adjustments to both arms of Study 1

thresh_2d(thresh, 1, 2,
           xlab = "Adjustment in Study 1 LOR: 3 vs. 1",
           ylab = "Adjustment in Study 1 LOR: 4 vs. 1",
           xlim = c(-1.5, 0.5), ylim = c(-2, 14),
           ybreaks = seq(-2, 14, 2))

thresh_forest

Producing threshold forest plots

Description

This function produces threshold forest plots, overlaying the decision-invariant intervals on the data points and their confidence/credible intervals, as shown by Phillippo et al. (2018).
thresh_forest

Usage

thresh_forest(
    thresh,
    y,
    CI.lo,
    CI.hi,
    label,
    orderby = NULL,
    data = NULL,
    CI.title = "95% Confidence Interval",
    label.title = "",
    y.title = "Mean",
    II.title = "Invariant Interval",
    xlab = "",
    xlim = NULL,
    sigfig = 3,
    digits = NULL,
    reline = NULL,
    clinsig = NULL,
    cutoff = NULL,
    II.colw = rgb(0.72, 0.8, 0.93),
    II.cols = rgb(0.93, 0.72, 0.8),
    II.lwd = 8,
    CI.lwd = 1,
    pointsize = 4,
    fontsize = 12,
    xbreaks = NULL,
    add.columns = NULL,
    add.columns.title = NULL,
    add.columns.after = -1,
    add.columns.hjust = 0.5,
    add.columns.uline = TRUE,
    calcdim = display,
    display = TRUE
)

Arguments

thresh
    A thresh object, as produced by `nma_thresh`.

y
    Data points. Either a column of data, or a numeric vector.

CI.lo
    Confidence/credible interval lower limits. Either a column of data, or a numeric vector.

CI.hi
    Confidence/credible interval upper limits. Either a column of data, or a numeric vector.

label
    Row labels (for each data point). Either a column of data, or a character vector.

orderby
    Variable(s) to order the table rows by. Either a column or columns of data, or a vector. By default, the rows are not reordered. Further arguments and/or multi-
ple ordering columns may be passed to the function order by instead providing a list containing the arguments to order.

data A data frame containing the data points y, confidence/credible intervals (CI.lo, CI.hi), and row labels labels. If data is not provided, the above variables will be searched for in the calling environment.

CI.title Title for CI column, default "95% Confidence Interval".

label.title Character string giving the heading for the row labels column.

y.title Character string giving the heading for the data points column, default "Mean".

II.title Title for invariant interval column, default "Invariant Interval".

xlab Character string giving the label for the x-axis.

xlim Numeric vector (length 2) of lower and upper limits for the x-axis. If not set, tries to choose a sensible default.

sigfig Number of significant digits to display in the table. Default 3.

digits Number of decimal places to display in the table. Overrides sigfig if set.

repline x intercept of vertical reference line, if desired.

clinsig Set the clinical significance level. Rows are marked with a dagger if they have one or more thresholds less than this value. Not set by default.

cutoff A single numeric value or numeric vector pair. Thresholds larger in magnitude than this value, or lying outside this interval, will be cut off and marked as NT (no threshold). Not set by default.

II.colw Colour for "wide" invariant intervals.

II.cols Colour for "short" invariant intervals.

II.lwd Line width of invariant intervals. Default 8.

CI.lwd Line width of confidence/credible intervals. Default 1.

pointsize Point size for forest plot means. Default 4.

fontsize Base font size. Default 12.

xbreaks Position of tick marks on the x-axis as a numeric vector.

add.columns Data frame (or matrix, vector) of additional columns to add to table.

add.columns.title Optional titles for the additional columns, otherwise use names from provided data.

add.columns.after Which column to add the new columns after? Default adds the columns to the far right.

add.columns.hjust Vector of horizontal justifications for the new columns, from 0 (left) to 1 (right). Default centres every column.

add.columns.uleline Underline the header of the new columns? Default TRUE.

calcdim Logical, calculate suggested output dimensions for saving to pdf? Calculates output size when TRUE; saves time when FALSE.

display Logical, display the plot? Defaults to TRUE.
thresh_forest

Value

Displays the forest plot on the current plot device (if display = TRUE). Also returns invisibly the underlying gtable object, which can be further manipulated.

Examples

# Please see the vignette "Examples" for worked examples including use of
# this function, including more information on the brief code below.

vignette("Examples", package = "nmathresh")

### Contrast level thresholds for Thrombolytic treatments NMA
K <- 6  # Number of treatments

# Contrast design matrix is
X <- matrix(ncol = K-1, byrow = TRUE,
c(1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1),
ncol = K-1, byrow = TRUE
)

# Reconstruct hypothetical likelihood covariance matrix using NNLS
lik.cov <- recon_vcov(Thrombo.post.cov, prior.prec = .0001, X = X)

# Thresholds are then
thresh <- nma_thresh(mean.dk = Thrombo.post.summary$statistics[1:(K-1), "Mean"],
  lhood = lik.cov,
  post = Thrombo.post.cov,
  nmatype = "fixed",
  X = X,
  opt.max = FALSE)

# Get treatment codes for the contrasts with data
d.a <- d.b <- vector(length = nrow(X))
for (i in 1:nrow(X)){
  d.a[i] <- ifelse(any(X[i, ] == -1), which(X[i, ] == -1), 0) + 1
  d.b[i] <- ifelse(any(X[i, ] == 1), which(X[i, ] == 1), 0) + 1
}

# Transform from d_ab style contrast references into d[i] style from the full set of contrasts
# for easy indexing in R
d.i <- d_ab2i(d.a, d.b, K = 6)

# Create plot data
plotdat <- data.frame(lab = paste0(d.b, " vs. ", d.a),
  contr.mean = Thrombo.post.summary$statistics[d.i, "Mean"],
  CI2.5 = Thrombo.post.summary$quantiles[d.i, "2.5%"],
  CI97.5 = Thrombo.post.summary$quantiles[d.i, "97.5%"]
)
```r
# Plot
thresh_forest(thresh, contr.mean, CI2.5, CI97.5, label = lab, data = plotdat,
    label.title = "Contrast", xlab = "Log Odds Ratio", CI.title = "95% Credible Interval",
    xlim = c(-.3, .3), reline = 0, digits = 2)
```

---

**Thrombo.post.cov**  
*Posterior covariance matrix from Thrombolytics NMA*

**Description**  
The posterior covariance matrix of the basic treatment effect parameters.

**Usage**

Thrombo.post.cov

**Format**

An object of class matrix with 5 rows and 5 columns.

**Source**

Generated from WinBUGS output, using the WinBUGS code from Caldwell et al. (2005). See also vignette("Examples",package = "nmathresh").

**References**


**See Also**

Thrombo.post.summary

---

**Thrombo.post.summary**  
*Posterior summary from Thrombolytics NMA*

**Description**

A summary.mcmc object of the type produced by the coda package, containing the requisite posterior summary information on the variables dd, the contrasts of the treatment effect parameters.

**Usage**

Thrombo.post.summary
Format

A `summary.mcmc` object. The key components for our use are:

- **statistics**: Matrix containing the posterior summary statistics, with columns for Mean, SD, Naive SE, and Time-series SE (also known as the Monte-Carlo standard error)
- **quantiles**: Matrix containing the posterior 2.5%, 25%, 50%, 75%, and 97.5% quantiles

Source

Generated from WinBUGS output, using the WinBUGS code from Caldwell et al. (2005). See also `vignette("Examples", package = "nmathresh")`.

References


See Also

`summary.mcmc`, `Thrombo.post.cov`
Index

*Topic** datasets
  SocAnx.post.cov, 8
  SocAnx.post.summary, 9
  Thrombo.post.cov, 16
  Thrombo.post.summary, 16

  d_ab2i, 2
  d_i2ab(d_ab2i), 2

  get.int, 3

  nma_thresh, 4, 8, 10, 11, 13

  recon_vcov, 6, 6

  SocAnx.post.cov, 8, 10
  SocAnx.post.summary, 9, 9
  summary.mcmc, 10, 17

  thresh-class, 10
  thresh_2d, 11
  thresh_forest, 6, 12
  Thrombo.post.cov, 16, 17
  Thrombo.post.summary, 16, 16