Package: pcnetmeta (via r-universe)

October 16, 2026

Type Package

Title Patient-Centered Network Meta-Analysis

Version 2.8

Date 2022-08-29

Maintainer Lifeng Lin <lifenglin@arizona.edu>

Depends R (>= 2.14.0), rjags (>= 4-6)

Imports coda, grDevices, methods

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

Description Performs Bayesian arm-based network meta-analysis for datasets with binary, continuous, and count outcomes (Zhang et al., 2014 <doi:10.1177/1740774513498322>; Lin et al., 2017 <doi:10.18637/jss.v080.i05>).

License GPL (>= 2)

NeedsCompilation no

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

Config/pak/sysreqs jags

Repository https://cran.r-universe.dev

Date/Publication 2022-08-31 07:30:02 UTC

RemoteUrl https://github.com/cran/pcnetmeta

RemoteRef HEAD

RemoteSha ecc9ffd53a7b5843b685f87fbfd1145d31c58c7d

Contents

pcnetmeta-package 2
absolute.plot 3
contrast.plot 4
diabetes 5

dietaryfat 6
nma.ab.bin 7
nma.ab.cont 11
nma.ab.followup 15
nma.ab.py 19
nma.networkplot 23
parkinson 25
rank.prob 26
smoke 27

Index **28**

`pcnetmeta-package` *Patient-Centered Network Meta-Analysis*

Description

Provides functions to perform Bayesian arm-based network meta-analysis for datasets with binary, continuous, and count outcomes.

Details

Much effort in the current literature of network meta-analysis has been devoted to contrast-based methods, which focus on modeling the relative treatment effects, such as log odds ratios. In some situations, patient-centered treatment-specific parameters, such as absolute risks, may be preferred. For example, when the outcome is binary, consider two scenarios comparing treatments A and B according to one-year survival rates: (i) $\pi_A = 0.8$ vs. $\pi_B = 0.5$ and (ii) $\pi_A = 0.004$ vs. $\pi_B = 0.001$. Both scenarios yield an odds ratio of 4.0, but patients would prefer treatment A in scenario (i) more strongly than in scenario (ii). The contrast-based network meta-analysis requires external data sources or separate modeling to estimate treatment-specific parameters.

Alternatively, the arm-based network meta-analysis focuses on estimating treatment-specific parameters, and relative effects can be subsequently obtained. The arm-based models for binary outcomes are discussed in Salanti et al. (2008) and Zhang et al. (2014).

This package provides user-friendly functions for the arm-based network meta-analysis. The function `nma.ab.bin` uses the model in Zhang et al. (2014) for binary outcomes; it provides estimates for the absolute risk (AR), risk ratio (RR), risk difference (RD), odds ratio (OR), log RR, log OR, as well as rank probabilities. This package also handles continuous outcomes and count data. A detailed instruction of using this package is provided in Lin et al. (2017).

Parameter estimation in the arm-based Bayesian hierarchical models is performed through JAGS. Note that this package does not include a copy of JAGS library, so users must install JAGS separately. Please refer to the JAGS home page at `http://mcmc-jags.sourceforge.net/` for instructions on downloading and installing JAGS.

Author(s)

Lifeng Lin, Jing Zhang, and Haitao Chu

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

References

Dias S, Sutton AJ, Ades AE, Welton NJ (2013). "Evidence synthesis for decision making 2: a generalized linear modeling framework for pairwise and network meta-analysis of randomized controlled trials." *Medical Decision Making*, **33**(5), 607–617. <doi: 10.1177/0272989X12458724>

Lin L, Zhang J, Hodges JS, Chu H (2017). "Performing arm-based network meta-analysis in R with the pcnetmeta package." *Journal of Statistical Software*, **80**(5), 1–25. <doi: 10.18637/jss.v080.i05>

Lu G, Ades AE (2004). "Combination of direct and indirect evidence in mixed treatment comparisons." *Statistics in Medicine*, **23**(20), 3105–3124. <doi: 10.1002/sim.1875>

Salanti G, Higgins JPT, Ades AE, Ioannidis JPA (2008). "Evaluation of networks of randomized trials." *Statistical Methods in Medical Research* **17**(3), 279–301. <doi: 10.1177/0962280207080643>

Zhang J, Carlin BP, Neaton JD, Soon GG, Nie L, Kane R, Virnig BA, Chu H (2014). "Network meta-analysis of randomized clinical trials: Reporting the proper summaries." *Clinical Trials*, **11**(2), 246–262. <doi: 10.1177/1740774513498322>

`absolute.plot` *Plotting Treatment-Specific 95% Credible Intervals*

Description

Generates a plot of 95% credible intervals (CIs) for the treatment-specific effect sizes.

Usage

```
absolute.plot(nma.obj, alphabetic = TRUE, digits = 2,
              save = TRUE, width = 5, height, network.name)
```

Arguments

`nma.obj`	a list object obtained by function `nma.ab.bin`, `nma.ab.cont`, `nma.ab.py`, or `nma.ab.followup`.
`alphabetic`	a logical value indicating whether to sort the CIs according to treatment names alphabetically. The default is `TRUE`. If `FALSE`, the CIs are plotted in the order of treatment IDs specified in `nma.obj`.
`digits`	an integer specifying the number of decimal places to be used for the point estimates and 95% CIs. The default is 2.
`save`	a logical value indicating whether to save the plot as a .pdf file. If `FALSE`, a plot window will be displayed. The following arguments are only used when `save = TRUE`.
`width`	a positive number specifying the plot width. The default is 5.
`height`	a positive number specifying the plot height. The default is the treatment number minus 1.
`network.name`	a character string indicating the network name to be used for the produced .pdf file name.

Value

None.

Examples

```
data("smoke")
# increase n.iter to reach convergence
set.seed(1234)
nma.out <- nma.ab.bin(s.id, t.id, r, n, data = smoke,
  trtname = c("NC", "SH", "IC", "GC"), param= "AR",
  model = "het_cor", n.adapt = 1000, n.iter = 100, n.chains = 1)
absolute.plot(nma.out, save = FALSE)
#absolute.plot(nma.out)
absolute.plot(nma.out, alphabetic = FALSE, save = FALSE)
```

---

contrast.plot *Contrast Plot of Relative Effect Measures*

Description

Generates a plot that shows 95% credible intervals (CIs) for relative effect measures.

Usage

```
contrast.plot(nma.obj, effect.measure, reference, digits = 2,
              save = TRUE, width = 5, height, network.name)
```

Arguments

nma.obj	a list object obtained by function nma.ab.bin, nma.ab.cont, nma.ab.py, or nma.ab.followup.
effect.measure	a character string specifying the relative effect measure to be shown in the contrast plot. If nma.obj is obtained from nma.ab.bin, this argument can be "OR" (the default), "LOR", "RR", "LRR", or "RD"; it can be also "OR.med" or "LOR.med" if they are specified in nma.ab.bin with link = "logit". If nma.obj is obtained from nma.ab.cont, this argument should be "diff" (the default). If nma.obj is obtained from nma.ab.py or nma.ab.followup, this argument can be "ratio" (the default) or "logratio". Note that the specified effect measures (or its logarithm/exponential) must have been estimated in nma.obj.
reference	a character string specifying the reference treatment name to be compared against.
digits	an integer specifying the number of decimal places to be used for the point estimates and 95% CIs. The default is 2.
save	a logical value indicating whether to save the plot as a .pdf file. If FALSE, a plot window will be displayed. The following arguments are only used when save = TRUE.

width	a positive number specifying the plot width. The default is 5.
height	a positive number specifying the plot height. The default is the treatment number minus 1.
network.name	a character string indicating the network name to be used for the produced .pdf file name.

Value

None.

Examples

```
data("smoke")
# increase n.iter to reach convergence
set.seed(1234)
nma.out <- nma.ab.bin(s.id, t.id, r, n, data = smoke,
  trtname = c("NC", "SH", "IC", "GC"), param = "LOR",
  model = "het_cor", n.adapt = 1000, n.iter = 100, n.chains = 1)
contrast.plot(nma.out, save = FALSE)
#contrast.plot(nma.out)
```

---

diabetes *Network Meta-Ananlysis on Diabetes*

Description

An example of network meta-analysis for binary outcomes with follow-up times reported.

Usage

```
data("diabetes")
```

Format

A data frame containing 22 studies which compare a total of 6 treatments. The outcome is binary; the mean follow-up time in each study is collected.

`s.id` a numeric vector of study IDs.

`folup` a numeric vector of follow-up time (in years) for each study.

`t.id` a numeric vector of treatment IDs.

`r` a numeric vector of the total mortality in each treatment group in each study.

`n` a numeric vector of the total number of participants in each treatment group in each study.

Details

This network meta-analysis is reported by Elliott and Meyer (2007) to assess the effects of anti-hypertensive agents on incident diabetes. Treatment IDs represent 1) diuretic; 2) placebo; 3) β-blocker; 4) calcium-channel blocker (CCB); 5) angiotensin-converting-enzyme (ACE) inhibitor; and 6) angiotensin-receptor blocker (ARB).

Source

Elliott WJ, Meyer PM (2007). "Incident diabetes in clinical trials of antihypertensive drugs: a network meta-analysis." *Lancet*, **369**(9557), 201–207. <doi: 10.1016/S01406736(07)601081>

`dietaryfat` *Network Meta-Ananlysis on Dietary Fat*

Description

An example of network meta-analysis for binary outcomes with exposure times in person-years reported.

Usage

```
data("dietaryfat")
```

Format

A data frame containing 10 studies which compare 2 treatments. The total number of person-years at risk are reported for each treatment group in each study.

`s.id` a numeric vector of study IDs.

`t.id` a numeric vector of treatment IDs.

`py` a numeric vector of the total person-years in each treatment group in each study.

`r` a numeric vector of the total mortality in each treatment group in each study.

`n` a numeric vector of the total number of participants in each treatment group in each study.

Details

This network meta-analysis is reported by Hooper et al. (2000) to assess the effects of change in dietary fats on cardiovascular mortality. Treatment 1 is a control diet and treatment 2 is a reduced fat diet. Originally, study 2 compares three treatments, which include treatments 1 and 2, and another diet. Here, we treat these two different types of diet as the same treatment, but keep the treatment arms separate.

Source

Hooper L, Summerbell CD, Higgins JPT, Thompson RL, Clements G, Capps N, Davey Smith G, Riemersma R, Ebrahim S (2000). "Reduced or modified dietary fat for preventing cardiovascular disease." *The Cochrane Database of Systematic Reviews*, **2**, Art. No.: CD002137. <doi: 10.1002/14651858.CD002137>

nma.ab.bin *Arm-Based Network Meta-Analysis for Binary Outcomes*

Description

Performs the arm-based network meta-analysis proposed by Zhang et al. (2014), including estimating the absolute risk (AR), risk difference (RD), odds ratio (OR), risk ratio (RR), log odds ratio (LOR), and log risk ratio (LRR).

Usage

```
nma.ab.bin(s.id, t.id, event.n, total.n, data, trtname,
           param = c("AR", "LOR", "LRR", "RD", "rank.prob"),
           model = "het_cor", link = "probit",
           prior.type, a = 0.001, b = 0.001, c = 10,
           higher.better = FALSE, digits = 4, n.adapt = 5000,
           n.iter = 100000, n.burnin = floor(n.iter/2), n.chains = 3,
           n.thin = max(1, floor((n.iter - n.burnin)/100000)),
           conv.diag = FALSE, trace = NULL, dic = FALSE, postdens = FALSE,
           mcmc.samples = FALSE)
```

Arguments

s.id	a numeric or character vector specifying study ID, or the corresponding column name in the argument data.
t.id	a numeric or character vector specifying treatment ID, or the corresponding column name in the argument data.
event.n	a numeric vector of non-negative integers, specifying event number for a certain treatment in the corresponding study, or the corresponding column name in the argument data.
total.n	a numeric vector of positive integers, specifying total number of participants for a certain treatment in the corresponding study, or the corresponding column name in the argument data.
data	an optional data frame containing the dataset for network meta-analysis. If data is specified, the previous arguments, s.id, t.id, event.n, and total.n, should be specified as the corresponding column names in data; otherwise, the previous arguments use environment variables.
trtname	a vector of character strings specifying the treatment names for the corresponding treatment IDs according to their order in t.id. If not specified, t.id is used as treatment names.
param	a vector of character strings specifying the effect measures to be estimated. The default includes "AR" (absolute risk), "LOR" (log odds ratio), "LRR" (log risk ratio), "RD" (risk difference), "rank.prob" (treatment rank probability). "AR" is automatically added into param even if it is not specified. In addition to the default measures, "OR" (odds ratio) and "RR" (risk ratio) can be added into param.

If the logit link is used (link = "logit"), "OR.med" (overall median odds ratio), "LOR.med" (overall median log odds ratio), and "rank.prob.med" (treatment rank probability based on overall median odds ratio) can be also added; see "Details". If model is "hom_eqcor" or "het_eqcor", param can also include "\rho" (the common correlation coefficient between treatments).

model — a character string specifying which Bayesian hierarchical model to be applied in the arm-based network meta-analysis. This argument can be set as "hom_eqcor", "het_eqcor", or "het_cor" (the default). See "Details" for the models.

link — a character string specifying the link function in the Bayesian hierarchical model for binary outcomes. It can be either "probit" (the default) or "logit".

prior.type — prior distribution of variances and/or covariances of random effects. If model is "hom_eqcor" or "het_eqcor", it can be set as "unif" (uniform prior for standard deviations, the default) or "invgamma" (inverse gamma prior for variances). If model is "het_cor", prior.type can be "invwishart" (the default) or "chol". Specifying "invwishart" yields an inverse-Wishart prior for the variance-covariance matrix of random effects; by specifying "chol", non-informative priors are assigned to variance and correlation components using the separation strategy by Cholesky decomposition. See "Details".

a, b — positive numbers, specifying the shape and scale parameters of inverse gamma priors for variance(s) of random effects if using prior.type as "invgamma" for model "hom_eqcor" or "het_eqcor". The default values for both parameters are 0.001.

c — positive number, specifying the upper bound of uniform prior for standard deviation(s) of random effects if using prior.type as "unif" for model "hom_eqcor" or "het_eqcor". The default is 10.

higher.better — an optional logical value which needs to be specified when estimating the treatment rank probabilities (i.e., "rank.prob" is included in the argument param). TRUE indicates that a higher event rate implies a better treatment, and vice versa. The default is FALSE.

digits — a positive integer specifying the digits after the decimal point for the effect measure estimates. The default is 4.

n.adapt — the number of iterations for adaptation in Markov chain Monte Carlo (MCMC) algorithm. The default is 5,000. If a warning "adaptation incomplete" appears, users may increase n.adapt. This argument and the following n.iter, n.burnin, n.chains, and n.thin are passed to the functions in R package **rjags**.

n.iter — the total number of iterations in each MCMC chain. The default is 100,000.

n.burnin — the number of iterations for burn-in period. The default is n.iter/2.

n.chains — the number of MCMC chains. The default is 3.

n.thin — a positive integer specifying the thinning rate. The default is the thinning rate which yields no more than 100,000 iterations remaining in each chain.

conv.diag — a logical value indicating whether to perform MCMC convergence diagnostic. The default is FALSE. If TRUE, n.chains must be greater than 1, and a .txt file, which contains the point estimates of the potential scale reduction factor and their upper confidence limits (see Gelman and Rubin 1992), will be written in users' current working directory.

`trace`	a vector of character strings of effect measures. The character strings should be selected from those specified in `param` (except `"rank.prob"`), and trace plots of the specified effect measures will be saved in users' current working directory. The default is not drawing trace plots (`NULL`).
`dic`	a logical value indicating whether to calculate the deviance information criterion (DIC) value. The default is `FALSE`.
`postdens`	a logical value indicating whether to draw the posterior density plots for treatment-specific absolue risks (ARs). If `TRUE`, a .pdf file containing the density plot will be saved in users' current working directory. The default is `FALSE`.
`mcmc.samples`	a logical value indicating whether to save MCMC posterior samples in the output object. The default is `FALSE`.

Details

Suppose that a network meta-analysis collects I studies on K treatments, where each study investigates a subset of the K treatments. Label the studies from $i = 1$ to I and the treatments from $k = 1$ to K. Let T_i be the subset of the K treatments that is compared in the ith study. Also, in the ith study, let n_{ik} be the number of participants allocated to treatment group k ($k \in T_i$), and y_{ik} be the number of events. The arm-based model is constructed as (Zhang et al. 2014):

$$y_{ik} \sim Bin(n_{ik}, p_{ik}) \qquad k \in T_i;$$
$$\Phi^{-1}(p_{ik}) = \mu_k + \nu_{ik};$$
$$(\nu_{i1}, \nu_{i2}, \ldots, \nu_{iK})^T \sim N(\mathbf{0}, \mathbf{\Sigma}_K),$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function, and $\mathbf{\Sigma}_K$ is a $K \times K$ positive definite covariance matrix. The μ_k's are treatment-specific fixed effects, and the random effects ν_{ik} are correlated within each study with the covariance matrix $\mathbf{\Sigma}_K$. The marginal absolute risk of treatment k is $p_k = E[p_{ik}]$; other effect measures are calculated based on these absolute risks.

An unstructured covariance matrix $\mathbf{\Sigma}_K$ in the model above corresponds to `model = "het_cor"`. The inverse-Wishart prior can be assigned to $\mathbf{\Sigma}_K$. Alternatively, using the separation strategy by Cholesky decomposition (`prior.type = "chol"`), uniform priors $U(0, c)$ are assigned to the standard deviations in $\mathbf{\Sigma}_K$ and non-informative priors are assigned to the correlation components (Barnard et al., 2000; Lu and Ades, 2009; Wei and Higgins, 2013; Lin and Chu, 2018). Denote σ_k as the standard deviation of ν_{ik} and $\mathbf{D} = diag(\sigma_1, \ldots, \sigma_K)$, then the correlation matrix is $\mathbf{R}_K = \mathbf{D}^{-1}\mathbf{\Sigma}_K\mathbf{D}^{-1}$. If we assume that all of the off-diagonal elements in $\mathbf{R}_K$ are equal, say to ρ, then this model corresponds to `model = "het_eqcor"`. If we further assume the homogeneity of variances of the random effects, that is, $\sigma_k = \sigma$ for $k = 1, 2, \ldots, K$, then the model is `"hom_eqcor"`. In addition, for the models `"hom_eqcor"` and `"het_eqcor"`, setting `prior.type` as `"invgamma"` implies using inverse-gamma priors with shape and scale parameters, a and b, for σ_k^2 or σ^2, and `"unif"` implies uniform priors $U(0, c)$ for σ_k or σ.

In addition to the probit link as used in the model above, one may also use the logit link (Chu et al., 2012), which is adopted more commonly in contrast-based models. Using the logit link, μ_k represents the overall median log odds across studies; thus, $\mu_k - \mu_h$ is the overall median log odds ratio between treatments h and k, $\exp(\mu_k - \mu_h)$ is the overall median odds ratio, and they futher yield the rank probabilities. The contrast-based models usually report such overall median (log) odds ratios, rather than the marginal effect measures obtained from the arm-based models. When the argument `link` is `"logit"`, users can additionally specify `"LOR.med"`, `"OR.med"`, and `"rank.prob.med"` to obtain these results based on the overall median odds ratios.

Value

`nma.ab.bin` returns a list with estimates of effect measures specified in `param`. If the argument `dic = TRUE`, the deviance information criterion (DIC) statistic will be returned in the output list. In addition, if `conv.diag = TRUE`, a .txt file containing the point estimates of the potential scale reduction factor and their upper confidence limits by Gelman and Rubin (1992) will be saved in users' current working directory. If `postdens = TRUE`, the posterior densities of treatment-specific absolute risks will be saved as a .pdf file. If `trace` is specified, the trace plots are saved as .png files.

Note

If there exists a treatment that has no event in all studies, errors may occur. To avoid such errors, users may set the zero event as a small positive number (say 0.5).

Earlier versions (< 4.0.0) of JAGS do not guarantee exact reproducibility of the results. We recommend users to install the latest version (>= 4.0.0) of JAGS so that exact reproducibility can be ensured by specifying certain seeds.

References

Barnard J, McCulloch R, Meng XL (2000). "Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage." *Statistica Sinica*, **10**(4), 1281–1311.

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

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

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

Lin L, Zhang J, Hodges JS, Chu H (2017). "Performing arm-based network meta-analysis in R with the pcnetmeta package." *Journal of Statistical Software*, **80**(5), 1–25. <doi: 10.18637/jss.v080.i05>

Lu G, Ades AE (2004). "Combination of direct and indirect evidence in mixed treatment comparisons." *Statistics in Medicine*, **23**(20), 3105–3124. <doi: 10.1002/sim.1875>

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

Spiegelhalter DJ, Best NG, Carlin BP, Van Der Linde A (2002). "Bayesian measures of model complexity and fit." *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, **64**(4), 583–639. <doi: 10.1111/14679868.00353>

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

Zhang J, Carlin BP, Neaton JD, Soon GG, Nie L, Kane R, Virnig BA, Chu H (2014). "Network meta-analysis of randomized clinical trials: Reporting the proper summaries." *Clinical Trials*, **11**(2), 246–262. <doi: 10.1177/1740774513498322>

See Also

`nma.ab.cont`, `nma.ab.py`, `nma.ab.followup`

Examples

```
data("smoke")

# For the smoke cessation data,
# higher event rate indicates better treatment

# use the model = "het_cor"
#set.seed(1234)
#het.cor.out <- nma.ab.bin(s.id, t.id, r, n, data = smoke,
#  trtname = c("NC", "SH", "IC", "GC"), param = c("AR", "OR", "RR", "LOR",
#  "LRR", "RD", "rank.prob"), model = "het_cor", higher.better = TRUE,
#  n.iter = 200000, n.thin = 1, conv.diag = TRUE, dic = TRUE,
#  trace = c("AR", "LOR"), postdens = TRUE)

# use the model = "hom_eqcor"
# increase n.iter to reach convergence
set.seed(123)
hom.eqcor.out <- nma.ab.bin(s.id, t.id, r, n, data = smoke,
  param = c("AR", "LRR"), model = "hom_eqcor", prior.type = "unif", c = 10,
  higher.better = TRUE, n.adapt = 1000, n.iter = 100, n.chains = 1)

# use the logit link
set.seed(1234)
hom.eqcor.out2 <- nma.ab.bin(s.id, t.id, r, n, data = smoke,
  param = c("AR", "OR", "OR.med", "rank.prob", "rank.prob.med"),
  model = "hom_eqcor", link = "logit", prior.type = "unif", c = 10,
  higher.better = TRUE, n.adapt = 1000, n.iter = 100, n.chains = 1)
```

---

nma.ab.cont *Arm-Based Network Meta-Analysis for Continuous Outcomes*

Description

Performs the arm-based network meta-analysis for continuous outcomes, which estimates the treatment-specific effects and effect differences.

Usage

```
nma.ab.cont(s.id, t.id, mean, sd, total.n, data, trtname,
            param = c("mu", "diff", "rank.prob"),model = "het_cor",
            prior.type, a = 0.001, b = 0.001, c = 10,
            higher.better = FALSE, digits = 4, n.adapt = 5000,
            n.iter = 100000, n.burnin = floor(n.iter/2), n.chains = 3,
            n.thin = max(1, floor((n.iter - n.burnin)/100000)),
            conv.diag = FALSE, trace = NULL, dic = FALSE, postdens = FALSE,
            mcmc.samples = FALSE)
```

Arguments

s.id	a numeric or character vector specifying study ID, or the corresponding column name in the argument data.
t.id	a numeric or character vector specifying treatment ID, or the corresponding column name in the argument data.
mean	a numeric vector specifying the sample mean of the continuous outcomes for a certain treatment group in the corresponding study, or the corresponding column name in the argument data.
sd	a numeric vector specifying the sample standard deviation of the continuous outcomes for a certain treatment group in the corresponding study, or the corresponding column name in the argument data.
total.n	a numeric vector of positive integers, indicating total number of participants for a certain treatment in the corresponding study, or the corresponding column name in the argument data.
data	an optional data frame containing the dataset of the network meta-analysis. If data is specified, the previous arguments, s.id, t.id, mean, sd, and total.n, should be specified as the corresponding column names in data; otherwise, the previous arguments use environment variables.
trtname	a vector of character strings specifying the treatment names for the corresponding treatment IDs according to their order in t.id. If not specified, t.id is used as treatment names.
param	a vector of character strings specifying the effect measures to be estimated. The default includes treatment-specific effects ("mu"), effect differences ("diff"), and treatment rank probabilities ("rank.prob"). "mu" is automatically added into param even if it is not specified.
model	a character string specifying which Bayesian hierarchical model to be applied in the arm-based network meta-analysis. This argument can be set as "hom_eqcor", "het_eqcor", or "het_cor" (the default). See "Details" for the models.
prior.type	prior distribution of variances and/or covariances of random effects. If model is "hom_eqcor" or "het_eqcor", it can be set as "unif" (uniform prior for standard deviations, the default) or "invgamma" (inverse gamma prior for variances). If model is "het_cor", prior.type can be "invwishart" (the default) or "chol". Specifying "invwishart" yields an inverse-Wishart prior for the variance-covariance matrix of random effects; by specifying "chol", non-informative priors are assigned to variance and correlation components using the separation strategy by Cholesky decomposition. See "Details".
a, b	positive numbers, specifying the shape and scale parameters of inverse gamma priors for variance(s) of random effects if using prior.type as "invgamma" for model "hom_eqcor" or "het_eqcor". The default values for both parameters are 0.001.
c	positive number, specifying the upper bound of uniform prior for standard deviation(s) of random effects if using prior.type as "unif" for model "hom_eqcor" or "het_eqcor". The default is 10.

`higher.better` an optional logical value which needs to be specified when estimating the treatment rank probabilities (i.e., `"rank.prob"` is included in the argument `param`). `TRUE` indicates that a higher sample mean of the continuous outcomes implies a better treatment, and vice versa. The default is `FALSE`.

`digits` a positive integer specifying the digits after the decimal point for the effect measure estimates. The default is 4.

`n.adapt` the number of iterations for adaptation in Markov chain Monte Carlo (MCMC) algorithm. The default is 5,000. If a warning "adaptation incomplete" appears, users may increase `n.adapt`. This argument and the following `n.iter`, `n.burnin`, `n.chains`, and `n.thin` are passed to the functions in R package **rjags**.

`n.iter` the total number of iterations in each MCMC chain. The default is 100,000.

`n.burnin` the number of iterations for burn-in period. The default is `n.iter/2`.

`n.chains` the number of MCMC chains. The default is 3.

`n.thin` a positive integer specifying the thinning rate. The default is the thinning rate which yields no more than 100,000 iterations remaining in each chain.

`conv.diag` a logical value indicating whether to perform MCMC convergence diagnostic. The default is `FALSE`. If `TRUE`, `n.chains` must be greater than 1, and a .txt file, which contains the point estimates of the potential scale reduction factor and their upper confidence limits (see Gelman and Rubin 1992), will be written in users' current working directory.

`trace` a vector of character strings of effect measures. The character strings should be selected from those specified in `param` (except `"rank.prob"`), and trace plots of the specified effect measures will be saved in users' current working directory. The default is not drawing trace plots (`NULL`).

`dic` a logical value indicating whether to calculate the deviance information criterion (DIC) value. The default is `FALSE`.

`postdens` a logical value indicating whether to draw the posterior density plots for treatment-specific effects. If `TRUE`, a .pdf file containing the density plot will be saved in users' current working directory. The default is `FALSE`.

`mcmc.samples` a logical value indicating whether to save MCMC posterior samples in the output object. The default is `FALSE`.

Details

Suppose that a network meta-analysis collects I studies on K treatments, where each study investigates a subset of the K treatments with continuous outcomes. Label the studies from $i = 1$ to I and the treatments from $k = 1$ to K. Let T_i be the subset of the K treatments that is compared in the ith study. Also, in the ith study, let n_{ik} be the number of participants allocated to treatment group k ($k \in T_i$), and $\bar{y}_{ik}$ and s^2_{ik} be the sample mean and sample standard deviation of the continuous outcome measures. The arm-based model is constructed as:

$$\bar{y}_{ik} \sim N(\theta_{ik}, s^2_{ik}/n_{ik}) \qquad k \in T_i;$$

$$\theta_{ik} = \mu_k + \nu_{ik};$$

$$(\nu_{i1}, \nu_{i2}, \ldots, \nu_{iK})^T \sim N(\mathbf{0}, \mathbf{\Sigma}_K),$$

where $\mathbf{\Sigma}_K$ is a $K \times K$ positive definite correlation matrix. The μ_k's are treatment-specific fixed effects, and the random effects ν_{ik} are correlated within each study with the covariance matrix $\mathbf{\Sigma}_K$.

An unstructured covariance matrix $\mathbf{\Sigma}_K$ in the model above corresponds to `model = "het_cor"`. The inverse-Wishart prior can be assigned to $\mathbf{\Sigma}_K$. Alternatively, using the separation strategy by Cholesky decomposition (`prior.type = "chol"`), uniform priors $U(0, c)$ are assigned to the standard deviations in $\mathbf{\Sigma}_K$ and non-informative priors are assigned to the correlation components (Barnard et al., 2000; Lu and Ades, 2009; Wei and Higgins, 2013; Lin and Chu, 2018). Denote σ_k as the standard deviation of ν_{ik} and $\mathbf{D} = diag(\sigma_1, \ldots, \sigma_K)$, then the correlation matrix is $\mathbf{R}_K = \mathbf{D}^{-1}\mathbf{\Sigma}_K\mathbf{D}^{-1}$. If we assume that all of the off-diagonal elements in $\mathbf{R}_K$ are equal, say to ρ, then this model corresponds to `model = "het_eqcor"`. If we further assume the homogeneity of variances of the random effects, that is, $\sigma_k = \sigma$ for $k = 1, 2, \ldots, K$, then the model is `"hom_eqcor"`. In addition, for the models `"hom_eqcor"` and `"het_eqcor"`, setting `prior.type` as `"invgamma"` implies using inverse-gamma priors with shape and scale parameters, a and b, for σ_k^2 or σ^2, and `"unif"` implies uniform priors $U(0, c)$ for σ_k or σ.

Value

`nma.ab.cont` returns a list with estimates of effect measures specified in `param`. If the argument `dic = TRUE`, the deviance information criterion (DIC) statistic will be returned in the output list. In addition, if `conv.diag = TRUE`, a .txt file containing the point estimates of the potential scale reduction factor and their upper confidence limits by Gelman and Rubin (1992) will be saved in users' current working directory. If `postdens = TRUE`, the posterior densities of treatment-specific absolute risks will be saved as a .pdf file. If `trace` is specified, the trace plots are saved as .png files.

Note

Earlier versions (< 4.0.0) of JAGS do not guarantee exact reproducibility of the results. We recommend users to install the latest version (>= 4.0.0) of JAGS so that exact reproducibility can be ensured by specifying certain seeds.

References

Barnard J, McCulloch R, Meng XL (2000). "Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage." *Statistica Sinica*, **10**(4), 1281–1311.

Dias S, Sutton AJ, Ades AE, Welton NJ (2013). "Evidence synthesis for decision making 2: a generalized linear modeling framework for pairwise and network meta-analysis of randomized controlled trials." *Medical Decision Making*, **33**(5), 607–617. <doi: 10.1177/0272989X12458724>

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

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

Lin L, Zhang J, Hodges JS, Chu H (2017). "Performing arm-based network meta-analysis in R with the pcnetmeta package." *Journal of Statistical Software*, **80**(5), 1–25. <doi: 10.18637/jss.v080.i05>

Lu G, Ades AE (2004). "Combination of direct and indirect evidence in mixed treatment comparisons." *Statistics in Medicine*, **23**(20), 3105–3124. <doi: 10.1002/sim.1875>

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

Spiegelhalter DJ, Best NG, Carlin BP, Van Der Linde A (2002). "Bayesian measures of model complexity and fit." *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, **64**(4), 583–639. <doi: 10.1111/14679868.00353>

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

Zhang J, Carlin BP, Neaton JD, Soon GG, Nie L, Kane R, Virnig BA, Chu H (2014). "Network meta-analysis of randomized clinical trials: Reporting the proper summaries." *Clinical Trials*, **11**(2), 246–262. <doi: 10.1177/1740774513498322>

See Also

nma.ab.bin, nma.ab.py, nma.ab.followup

Examples

```
data("parkinson")
# increase n.iter to reach convergence of MCMC
# increase n.adapt to enhance efficiency
set.seed(1234)
cont.out <- nma.ab.cont(s.id, t.id, mean, sd, n, data = parkinson,
  param = c("mu", "diff"), model = "hom_eqcor", prior.type = "unif",
  n.adapt = 200, n.iter = 100, n.chains = 1)
```

---

nma.ab.followup *Arm-Based Network Meta-Analysis for Binary Outcomes with Follow-Up Times Reported*

Description

Performs the arm-based network meta-analysis for binary outcomes when the follow-up time of each study is reported, which estimates the treatment-specific rate, rate ratio between treatments, and their logarithms.

Usage

```
nma.ab.followup(s.id, t.id, event.n, total.n, followup, data, trtname,
                param = c("lograte", "logratio", "rank.prob"),
                model = "het_cor", prior.type, a = 0.001, b = 0.001,
                c = 10, higher.better = FALSE, digits = 4, n.adapt = 5000,
                n.iter = 100000, n.burnin = floor(n.iter/2), n.chains = 3,
                n.thin = max(1, floor((n.iter - n.burnin)/100000)),
                conv.diag = FALSE, trace = NULL, dic = FALSE,
                postdens = FALSE, mcmc.samples = FALSE)
```

Arguments

s.id	a numeric or character vector specifying study ID, or the corresponding column name in the argument data.
t.id	a numeric or character vector specifying treatment ID, or the corresponding column name in the argument data.
event.n	a numeric vector of non-negative integers, specifying number of events in each study's treatment group, or the corresponding column name in the argument data.
total.n	a numeric vector of non-negative integers, specifying total number of participants in each study's treatment group, or the corresponding column name in the argument data.
followup	a numeric vector of positive numbers, specifying follow-up times for different studies, or the corresponding column name in the argument data.
data	an optional data frame containing the dataset of the network meta-analysis. If data is specified, the previous arguments, s.id, t.id, event.n, total.n, and followup, should be specified as the corresponding column names in data; otherwise, the previous arguments use environment variables.
trtname	a vector of character strings specifying the treatment names for the corresponding treatment IDs according to their order in t.id. If not specified, t.id is used as treatment names.
param	a vector of character strings specifying the effect measures to be estimated. The default includes log treatment-specific rate ("lograte"), log rate ratio ("logratio"), and treatment rank probability ("rank.prob"). "lograte" is automatically added into param even if it is not specified. In addition to the default measures, treatment-specific rate ("rate") and rate ratio ("ratio") can be added into the argument param.
model	a character string specifying which Bayesian hierarchical model to be applied in the arm-based network meta-analysis. This argument can be set as "hom_eqcor", "het_eqcor", or "het_cor" (the default). See "Details" for the models.
prior.type	prior distribution of variances and/or covariances of random effects. If model is "hom_eqcor" or "het_eqcor", it can be set as "unif" (uniform prior for standard deviations, the default) or "invgamma" (inverse gamma prior for variances). If model is "het_cor", prior.type can be "invwishart" (the default) or "chol". Specifying "invwishart" yields an inverse-Wishart prior for the variance-covariance matrix of random effects; by specifying "chol", non-informative priors are assigned to variance and correlation components using the separation strategy by Cholesky decomposition. See "Details".
a, b	positive numbers, specifying the shape and scale parameters of inverse gamma priors for variance(s) of random effects if using prior.type as "invgamma" for model "hom_eqcor" or "het_eqcor". The default values for both parameters are 0.001.
c	positive number, specifying the upper bound of uniform prior for standard deviation(s) of random effects if using prior.type as "unif" for model "hom_eqcor" or "het_eqcor". The default is 10.

`higher.better` an optional logical value which needs to be specified when estimating the treatment rank probabilities (i.e., "rank.prob" is included in argument `param`). `TRUE` indicates that a higher treatment-specific rate implies a better treatment, and vice versa. The default is `FALSE`.

`digits` a positive integer specifying the digits after the decimal point for the effect measure estimates. The default is 4.

`n.adapt` the number of iterations for adaptation in Markov chain Monte Carlo (MCMC) algorithm. The default is 5,000. If a warning "adaptation incomplete" appears, users may increase `n.adapt`. This argument and the following `n.iter`, `n.burnin`, `n.chains`, and `n.thin` are passed to the functions in R package **rjags**.

`n.iter` the total number of iterations in each MCMC chain. The default is 100,000.

`n.burnin` the number of iterations for burn-in period. The default is `n.iter/2`.

`n.chains` the number of MCMC chains. The default is 3.

`n.thin` a positive integer specifying the thinning rate. The default is the thinning rate which yields no more than 100,000 iterations remaining in each chain.

`conv.diag` a logical value indicating whether to perform MCMC convergence diagnostic. The default is `FALSE`. If `TRUE`, `n.chains` must be greater than 1, and a .txt file, which contains the point estimates of the potential scale reduction factor and their upper confidence limits (see Gelman and Rubin 1992), will be written in users' current working directory.

`trace` a vector of character strings of effect measures. The character strings should be selected from those specified in `param` (except `"rank.prob"`), and trace plots of the specified effect measures will be saved in users' current working directory. The default is not drawing trace plots (`NULL`).

`dic` a logical value indicating whether to calculate the deviance information criterion (DIC) value. The default is `FALSE`.

`postdens` a logical value indicating whether to draw the posterior density plots for treatment-specific rates. If `TRUE`, a .pdf file containing the density plot will be saved in users' current working directory. The default is `FALSE`.

`mcmc.samples` a logical value indicating whether to save MCMC posterior samples in the output object. The default is `FALSE`.

Details

Suppose that a network meta-analysis collects I studies on K treatments, where each study investigates a subset of the K treatments. The outcome is binary, and the follow-up time for each study is reported. Label the studies from $i = 1$ to I and the treatments from $k = 1$ to K. Let T_i be the subset of the K treatments that is compared in the ith study. Also, in the ith study, let y_{ik} and n_{ik} be the number of events and the total number of participants in treatment group k. Denote f_i as the follow-up time of the ith study. The arm-based network meta-analysis model for these settings is constructed as:

$$y_{ik} \sim Bin(n_{ik}, p_{ik}) \qquad k \in T_i;$$
$$cloglog(p_{ik}) = \log(f_i) + \log(\lambda_{ik});$$
$$\log(\lambda_{ik}) = \mu_k + \nu_{ik};$$

$$(\nu_{i1}, \nu_{i2}, \ldots, \nu_{iK})^T \sim N(\mathbf{0}, \mathbf{\Sigma}_K),$$

where $cloglot(t) = \log(-\log(1-t))$ is the complementary log-log link function, and $\mathbf{\Sigma}_K$ is a $K \times K$ positive definite covariance matrix. The μ_k's are treatment-specific fixed effects, and the random effects ν_{ik} are correlated within each study with the covariance matrix $\mathbf{\Sigma}_K$.

An unstructured covariance matrix $\mathbf{\Sigma}_K$ in the model above corresponds to `model = "het_cor"`. The inverse-Wishart prior can be assigned to $\mathbf{\Sigma}_K$. Alternatively, using the separation strategy by Cholesky decomposition (`prior.type = "chol"`), uniform priors $U(0, c)$ are assigned to the standard deviations in $\mathbf{\Sigma}_K$ and non-informative priors are assigned to the correlation components (Barnard et al., 2000; Lu and Ades, 2009; Wei and Higgins, 2013; Lin and Chu, 2018). Denote σ_k as the standard deviation of ν_{ik} and $\mathbf{D} = diag(\sigma_1, \ldots, \sigma_K)$, then the correlation matrix is $\mathbf{R}_K = \mathbf{D}^{-1}\mathbf{\Sigma}_K\mathbf{D}^{-1}$. If we assume that all of the off-diagonal elements in $\mathbf{R}_K$ are equal, say to ρ, then this model corresponds to `model = "het_eqcor"`. If we further assume the homogeneity of variances of the random effects, that is, $\sigma_k = \sigma$ for $k = 1, 2, \ldots, K$, then the model is `"hom_eqcor"`. In addition, for the models `"hom_eqcor"` and `"het_eqcor"`, setting `prior.type` as `"invgamma"` implies using inverse-gamma priors with shape and scale parameters, a and b, for σ_k^2 or σ^2, and `"unif"` implies uniform priors $U(0, c)$ for σ_k or σ.

Value

`nma.ab.followup` returns a list with estimates of effect measures specified in `param`. If the argument `dic = TRUE`, the deviance information criterion (DIC) statistic will be returned in the output list. In addition, if `conv.diag = TRUE`, a .txt file containing the point estimates of the potential scale reduction factor and their upper confidence limits by Gelman and Rubin (1992) will be saved in users' current working directory. If `postdens = TRUE`, the posterior densities of treatment-specific absolute risks will be saved as a .pdf file. If `trace` is specified, the trace plots are saved as .png files.

Note

Earlier versions (< 4.0.0) of JAGS do not guarantee exact reproducibility of the results. We recommend users to install the latest version (>= 4.0.0) of JAGS so that exact reproducibility can be ensured by specifying certain seeds.

References

Barnard J, McCulloch R, Meng XL (2000). "Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage." *Statistica Sinica*, **10**(4), 1281–1311.

Dias S, Sutton AJ, Ades AE, Welton NJ (2013). "Evidence synthesis for decision making 2: a generalized linear modeling framework for pairwise and network meta-analysis of randomized controlled trials." *Medical Decision Making*, **33**(5), 607–617. <doi: 10.1177/0272989X12458724>

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

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

Lin L, Zhang J, Hodges JS, Chu H (2017). "Performing arm-based network meta-analysis in R with the pcnetmeta package." *Journal of Statistical Software*, **80**(5), 1–25. <doi: 10.18637/jss.v080.i05>

Lu G, Ades AE (2004). "Combination of direct and indirect evidence in mixed treatment comparisons." *Statistics in Medicine*, **23**(20), 3105–3124. <doi: 10.1002/sim.1875>

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

Spiegelhalter DJ, Best NG, Carlin BP, Van Der Linde A (2002). "Bayesian measures of model complexity and fit." *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, **64**(4), 583–639. <doi: 10.1111/14679868.00353>

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

Zhang J, Carlin BP, Neaton JD, Soon GG, Nie L, Kane R, Virnig BA, Chu H (2014). "Network meta-analysis of randomized clinical trials: Reporting the proper summaries." *Clinical Trials*, **11**(2), 246–262. <doi: 10.1177/1740774513498322>

See Also

nma.ab.bin, nma.ab.cont, nma.ab.py

Examples

```
#data("diabetes")
# increase n.iter to reach convergence of MCMC
# increase n.adapt to enhance efficiency
#set.seed(1234)
#followup.out <- nma.ab.followup(s.id, t.id, r, n, folup, data = diabetes,
#  model = "het_cor", n.adapt = 500, n.iter = 100, n.chains = 1)
```

---

nma.ab.py *Arm-Based Network Meta-Analysis for Count Datasets with Exposure Time in Person-Years Reported*

Description

Performs the arm-based network meta-analysis for count datasets when the exposure times (in person-years) are reported, which estimates the treatment-specific rate, rate ratio between treatments, and their logarithms.

Usage

```
nma.ab.py(s.id, t.id, event.n, py, data, trtname,
          param = c("lograte", "logratio", "rank.prob"), model = "het_cor",
          prior.type, a = 0.001, b = 0.001, c = 10, higher.better = FALSE,
          digits = 4, n.adapt = 5000, n.iter = 100000,
          n.burnin = floor(n.iter/2), n.chains = 3,
          n.thin = max(1, floor((n.iter - n.burnin)/100000)),
          conv.diag = FALSE, trace = NULL, dic = FALSE, postdens = FALSE,
          mcmc.samples = FALSE)
```

Arguments

s.id	a numeric or character vector specifying study ID, or the corresponding column name in the argument data.
t.id	a numeric or character vector specifying treatment ID, or the corresponding column name in the argument data.
event.n	a numeric vector of non-negative integers, specifying number of events in each study's treatment group, or the corresponding column name in the argument data.
py	a numeric vector of non-negative numbers, specifying exposure time in person-years in each study's treatment group, or the corresponding column name in the argument data.
data	an optional data frame containing the dataset of the network meta-analysis. If data is specified, the previous arguments, s.id, t.id, event.n, and py, should be specified as the corresponding column names in data; otherwise, the previous arguments use environment variables.
trtname	a vector of character strings specifying the treatment names for the corresponding treatment IDs according to their order in t.id. If not specified, t.id is used as treatment names.
param	a vector of character strings specifying the effect measures to be estimated. The default includes log treatment-specific rate ("lograte"), log rate ratio ("logratio"), and treatment rank probability ("rank.prob"). "lograte" is automatically added into param even if it is not specified. In addition to the default measures, treatment-specific rate ("rate") and rate ratio ("ratio") can be added into the argument param.
model	a character string specifying which Bayesian hierarchical model to be applied in the arm-based network meta-analysis. This argument can be set as "hom_eqcor", "het_eqcor", or "het_cor" (the default). See "Details" for the models.
prior.type	prior distribution of variances and/or covariances of random effects. If model is "hom_eqcor" or "het_eqcor", it can be set as "unif" (uniform prior for standard deviations, the default) or "invgamma" (inverse gamma prior for variances). If model is "het_cor", prior.type can be "invwishart" (the default) or "chol". Specifying "invwishart" yields an inverse-Wishart prior for the variance-covariance matrix of random effects; by specifying "chol", non-informative priors are assigned to variance and correlation components using the separation strategy by Cholesky decomposition. See "Details".
a, b	positive numbers, specifying the shape and scale parameters of inverse gamma priors for variance(s) of random effects if using prior.type as "invgamma" for model "hom_eqcor" or "het_eqcor". The default values for both parameters are 0.001.
c	positive number, specifying the upper bound of uniform prior for standard deviation(s) of random effects if using prior.type as "unif" for model "hom_eqcor" or "het_eqcor". The default is 10.
higher.better	an optional logical value which needs to be specified when estimating the treatment rank probabilities (i.e., "rank.prob" is included in argument param). TRUE indicates that a higher treatment-specific rate implies a better treatment, and vice versa. The default is FALSE.

`digits` a positive integer specifying the digits after the decimal point of the effect measures estimates. The default is 4.

`n.adapt` the number of iterations for adaptation in Markov chain Monte Carlo (MCMC) algorithm. The default is 5,000. If a warning "adaptation incomplete" appears, users may increase `n.adapt`. This argument and the following `n.iter`, `n.burnin`, `n.chains`, and `n.thin` are passed to the functions in R package **rjags**.

`n.iter` the total number of iterations in each MCMC chain. The default is 100,000.

`n.burnin` the number of iterations for burn-in period. The default is `n.iter/2`.

`n.chains` the number of MCMC chains. The default is 3.

`n.thin` a positive integer specifying the thinning rate. The default is the thinning rate which yields no more than 100,000 iterations remaining in each chain.

`conv.diag` a logical value indicating whether to perform MCMC convergence diagnostic. The default is `FALSE`. If `TRUE`, `n.chains` must be greater than 1, and a .txt file, which contains the point estimates of the potential scale reduction factor and their upper confidence limits (see Gelman and Rubin 1992), will be written in users' current working directory.

`trace` a vector of character strings of effect measures. The character strings should be selected from those specified in `param` (except `"rank.prob"`), and trace plots of the specified effect measures will be saved in users' current working directory. The default is not drawing trace plots (`NULL`).

`dic` a logical value indicating whether to calculate the deviance information criterion (DIC) value. The default is `FALSE`.

`postdens` a logical value indicating whether to draw the posterior density plots for treatment-specific rates. If `TRUE`, a .pdf file containing the density plot will be saved in users' current working directory. The default is `FALSE`.

`mcmc.samples` a logical value indicating whether to save MCMC posterior samples in the output object. The default is `FALSE`.

Details

Suppose that a network meta-analysis collects I studies on K treatments, where each study investigates a subset of the K treatments. The exposure time in person-years and the count of events in each treatment group are reported. Label the studies from $i = 1$ to I and the treatments from $k = 1$ to K. Let T_i be the subset of the K treatments that is compared in the ith study. Also, in the ith study, let y_{ik} be the number of events in treatment group k, and E_{ik} be the corresponding exposure time in person-years. The arm-based network meta-analysis model for these settings is constructed as:

$$y_{ik} \sim Pois(E_{ik}\lambda_{ik}) \qquad k \in T_i;$$

$$\log(\lambda_{ik}) = \mu_k + \nu_{ik};$$

$$(\nu_{i1}, \nu_{i2}, \ldots, \nu_{iK})^T \sim N(\mathbf{0}, \mathbf{\Sigma}_K),$$

where $\mathbf{\Sigma}_K$ is a $K \times K$ positive definite correlation matrix. The μ_k's are treatment-specific fixed effects, and the random effects ν_{ik} are correlated within each study with the covariance matrix $\mathbf{\Sigma}_K$.

An unstructured covariance matrix $\mathbf{\Sigma}_K$ in the model above corresponds to `model = "het_cor"`. The inverse-Wishart prior can be assigned to $\mathbf{\Sigma}_K$. Alternatively, using the separation strategy by Cholesky decomposition (`prior.type = "chol"`), uniform priors $U(0, c)$ are assigned to the standard deviations in $\mathbf{\Sigma}_K$ and non-informative priors are assigned to the correlation components (Barnard et al., 2000; Lu and Ades, 2009; Wei and Higgins, 2013; Lin and Chu, 2018). Denote σ_k as the standard deviation of ν_{ik} and $\mathbf{D} = diag(\sigma_1, \ldots, \sigma_K)$, then the correlation matrix is $\mathbf{R}_K = \mathbf{D}^{-1}\mathbf{\Sigma}_K\mathbf{D}^{-1}$. If we assume that all of the off-diagonal elements in $\mathbf{R}_K$ are equal, say to ρ, then this model corresponds to `model = "het_eqcor"`. If we further assume the homogeneity of variances of the random effects, that is, $\sigma_k = \sigma$ for $k = 1, 2, \ldots, K$, then the model is `"hom_eqcor"`. In addition, for the models `"hom_eqcor"` and `"het_eqcor"`, setting `prior.type` as `"invgamma"` implies using inverse-gamma priors with shape and scale parameters, a and b, for σ_k^2 or σ^2, and `"unif"` implies uniform priors $U(0, c)$ for σ_k or σ.

Value

`nma.ab.py` returns a list with estimates of effect measures specified in `param`. If the argument `dic = TRUE`, the deviance information criterion (DIC) statistic will be returned in the output list. In addition, if `conv.diag = TRUE`, a .txt file containing the point estimates of the potential scale reduction factor and their upper confidence limits by Gelman and Rubin (1992) will be saved in users' current working directory. If `postdens = TRUE`, the posterior densities of treatment-specific absolute risks will be saved as a .pdf file. If `trace` is specified, the trace plots are saved as .png files.

Note

Earlier versions (< 4.0.0) of JAGS do not guarantee exact reproducibility of the results. We recommend users to install the latest version (>= 4.0.0) of JAGS so that exact reproducibility can be ensured by specifying certain seeds.

References

Barnard J, McCulloch R, Meng XL (2000). "Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage." *Statistica Sinica*, **10**(4), 1281–1311.

Dias S, Sutton AJ, Ades AE, Welton NJ (2013). "Evidence synthesis for decision making 2: a generalized linear modeling framework for pairwise and network meta-analysis of randomized controlled trials." *Medical Decision Making*, **33**(5), 607–617. <doi: 10.1177/0272989X12458724>

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

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

Lin L, Zhang J, Hodges JS, Chu H (2017). "Performing arm-based network meta-analysis in R with the pcnetmeta package." *Journal of Statistical Software*, **80**(5), 1–25. <doi: 10.18637/jss.v080.i05>

Lu G, Ades AE (2004). "Combination of direct and indirect evidence in mixed treatment comparisons." *Statistics in Medicine*, **23**(20), 3105–3124. <doi: 10.1002/sim.1875>

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

Spiegelhalter DJ, Best NG, Carlin BP, Van Der Linde A (2002). "Bayesian measures of model complexity and fit." *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, **64**(4), 583–639. <doi: 10.1111/14679868.00353>

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

Zhang J, Carlin BP, Neaton JD, Soon GG, Nie L, Kane R, Virnig BA, Chu H (2014). "Network meta-analysis of randomized clinical trials: Reporting the proper summaries." *Clinical Trials*, **11**(2), 246–262. <doi: 10.1177/1740774513498322>

See Also

nma.ab.bin, nma.ab.cont, nma.ab.followup

Examples

```
#data("dietaryfat")
# increase n.iter to reach convergence of MCMC
# increase n.adapt to enhance efficiency
#set.seed(1234)
#py.out <- nma.ab.py(s.id, t.id, r, py, data = dietaryfat, model = "het_cor",
#  n.adapt = 300, n.iter = 100, n.chains = 1)
```

---

nma.networkplot *Plotting Treatment Network*

Description

Visualizes the network of treatment comparisons.

Usage

```
nma.networkplot(s.id, t.id, n, data, title = "", title.cex = 1, trtname,
                alphabetic = TRUE, multi.show = FALSE, multi.col,
                weight.edge = TRUE, adjust.thick = 5, weight.node = TRUE,
                weight.node.ss = FALSE, adjust.node.size = 10,
                node.col = "gray", edge.col = "black", text.cex = 1,
                adjust.figsizex = 1.1, adjust.figsizey = 1.1)
```

Arguments

s.id	a numeric or character vector specifying study ID, or the corresponding column name in the argument data.
t.id	a numeric or character vector specifying treatment ID, or the corresponding column name in the argument data.
n	a numeric or character vector specifying sample size, or the corresponding column name in the argument data. It is used only when the node size is proportional to the total sample size of the corresponding treatment (weight.node.ss = TRUE).

data	an optional data frame containing the dataset of the network meta-analysis. If data is specified, the previous arguments, s.id, t.id, and n, should be specified as the corresponding column names in data; otherwise, the previous arguments use environment variables.
title	a character string indicating the plot title.
title.cex	the size of the plot title.
trtname	a vector of character strings specifying the treatment names for the corresponding treatment IDs according to their order in t.id. If not specified, t.id is used as treatment names.
alphabetic	a logical value indicating whether to sort treatment nodes alphabetically in the network plot. The default is TRUE. If FALSE, treatment nodes are sorted by the treatment IDs specified in t.id.
multi.show	a logical value indicating whether to show multi-arm studies (if any). Multi-arm studies with different treatment sets are visualized by shaded treatment loops in different colors.
multi.col	an optional vector of colors for visualizing multi-arm studies.
weight.edge	a logical value indicating whether to draw the edges proportionally to the number of direct treatment comparisons. The default is TRUE.
adjust.thick	a positive integer specifying the maximum thickness of the edge when weight is TRUE. The default is 5.
weight.node	a logical value indicating whether the node size is weighted by the total sample size or the total number of direct treatment comparisons of the corresponding treatment (TRUE, the default) or all treatment nodes have equal sizes (FALSE).
weight.node.ss	a logical value indicating whether the node size is proportional to the total sample size (TRUE) or the total number of direct treatment comparisons (FALSE, the default) of the corresponding treatment.
adjust.node.size	a positive number to adjust the node sizes when weight.node is TRUE. The default is 10.
node.col	a character string specifying the color of treatment nodes. The default is "gray".
edge.col	a character string specifying the color of edges between treatments nodes. The default is "black".
text.cex	a positive integer specifying the sizes of treatment names placed around/on the corresponding nodes. The default is 1.
adjust.figsizex	a positive number used to adjust the plot width. The default is 1.1.
adjust.figsizey	a positive number used to adjust the plot height. The default is 1.1.

Value

A network plot is generated. Each node represents a treatment, and the edges indicate the direct comparisons between the two treatments.

References

Lin L, Zhang J, Hodges JS, Chu H (2017). "Performing arm-based network meta-analysis in R with the pcnetmeta package." *Journal of Statistical Software*, **80**(5), 1–25. <doi: 10.18637/jss.v080.i05>

Examples

```
data("smoke")
# NC: No contact; SH: Self-help
# IC: individual counseling; GC: group counseling
nma.networkplot(s.id, t.id, data = smoke, title = "Smoke Cessation",
  trtname = c("NC", "SH", "IC", "GC"))
nma.networkplot(s.id, t.id, n, data = smoke, title = "Smoke Cessation",
  trtname = c("NC", "SH", "IC", "GC"),
  weight.node.ss = TRUE, multi.show = TRUE)

data("diabetes")
nma.networkplot(s.id, t.id, data = diabetes, title = "Diabetes",
  trtname = c("Diuretic", "Placebo", "b-blocker", "CCB", "ACE inhibitor",
  "ARB"))
nma.networkplot(s.id, t.id, data = diabetes, title = "Diabetes",
  trtname = c("Diuretic", "Placebo", "b-blocker", "CCB", "ACE inhibitor",
  "ARB"), multi.show = TRUE)
nma.networkplot(s.id, t.id, n, data = diabetes, title = "",
  weight.node.ss = TRUE, multi.show = TRUE)
mtext(text = "Diabetes", side = 3, line = 0, font = 1, cex = 2)
```

---

parkinson *Network Meta-Ananlysis on Parkinson's Disease*

Description

An example of network meta-analysis for continuous outcomes.

Usage

```
data("parkinson")
```

Format

A data frame containing 7 studies which compare 5 treatments.

`s.id` a numeric vector of study IDs.

`t.id` a numeric vector of treatment IDs.

`mean` a numeric vector of the mean of continuous outcomes in each treatment group in each study.

`sd` a numeric vector of the standard deviance of continuous outcomes in each treatment group in each study.

`n` a numeric vector of the total number of participants in each treatment group in each study.

Details

The continuous outcome measures the off-time reduction in patients given dopamine agonists as adjunct therapy in Parkinson's disease. Treatment 1 is placebo and treatments 2 to 5 are active drugs.

Source

Dias S, Sutton AJ, Ades AE, Welton NJ (2013). "Evidence synthesis for decision making 2: a generalized linear modeling framework for pairwise and network meta-analysis of randomized controlled trials." *Medical Decision Making*, **33**(5), 607–617. <doi: 10.1177/0272989X12458724>

`rank.prob` *Plotting Treatment Rank Probabilities*

Description

`rank.prob` generates a plot for treatment rank probabilities.

Usage

```
rank.prob(nma.obj, cex.axis = 1, cex.lab = 1, median = FALSE)
```

Arguments

`nma.obj`	a list object obtained by function `nma.ab.bin`, `nma.ab.cont`, `nma.ab.py`, or `nma.ab.followup`.
`cex.axis`	a numeric value specifying the size of the tick label numbers/text.
`cex.lab`	a numeric value specifying the size of the axis label text.
`median`	a logical value indicating whether the rank probabilities are based on the marginal effect measures (`FALSE`, the default) or the overall median odds ratios. This argument is only used when `link = "logit"` in `nma.ab.bin`.

Details

A plot of treatment rank probabilities is generated. Each vertical bar represents probabilities of being different ranks for a specific treatment. A darker area indicates the probability of being a higher rank. The black area indicates the probability of being the best treatment.

Value

None.

Examples

```
data("smoke")
# increase n.iter to reach convergence
set.seed(1234)
nma.out <- nma.ab.bin(s.id, t.id, r, n, data = smoke,
  trtname = c("NC", "SH", "IC", "GC"), param = "rank.prob", model = "het_cor",
  higher.better = TRUE, n.adapt = 1000, n.iter = 100, n.chains = 1)
rank.prob(nma.out)
```

---

smoke *Network Meta-Analysis on Smoking Cessation Data*

Description

An example of network meta-analysis for binary outcomes.

Usage

```
data("smoke")
```

Format

A data frame containing 24 studies on smoking cessation, comparing four treatments.

`s.id` a numeric vector of study IDs.

`t.id` a numeric vector of treatment IDs.

`r` a numeric vector of the event count for a certain treatment in the corresponding study.

`n` a numeric vector of the total number of participants for a certain treatment in the corresponding study.

Details

Treatment IDs stand for 1) no contact; 2) self-help; 3) individual counseling; and 4) group counseling.

Source

Hasselblad V (1998) "Meta-analysis of multitreatment studies." *Medical Decision Making*, **18**(1), 37–43. <doi: 10.1177/0272989X9801800110>

Lu G, Ades AE (2006) "Assessing evidence inconsistency in mixed treatment comparisons." *Journal of the American Statistical Association* **101**(474), 447–459. <doi: 10.1198/016214505000001302>

Index

∗ **dataset**
diabetes, 5
dietaryfat, 6
parkinson, 25
smoke, 27
∗ **network meta-analysis**
nma.ab.bin, 7
nma.ab.cont, 11
nma.ab.followup, 15
nma.ab.py, 19
∗ **plot**
absolute.plot, 3
contrast.plot, 4
nma.networkplot, 23
rank.prob, 26

absolute.plot, 3

contrast.plot, 4

diabetes, 5
dietaryfat, 6

nma.ab.bin, *2–4*, 7, *15*, *19*, *23*, *26*
nma.ab.cont, *3, 4*, *10*, 11, *19*, *23*, *26*
nma.ab.followup, *3, 4*, *10*, 15, *15*, *23*, *26*
nma.ab.py, *3, 4*, *10*, *15*, 19, *19*, *26*
nma.networkplot, 23

parkinson, 25
pcnetmeta *(*pcnetmeta-package*)*, 2
pcnetmeta-package, 2

rank.prob, 26

smoke, 27