

Journal of Statistical Software

March 2023, Volume 106, Issue 2. doi: 10.18637/jss.v106.i02

netmeta: An R Package for Network Meta-Analysis Using Frequentist Methods

Sara Balduzzi
Universität Freiburg

Gerta Rücker
Universität Freiburg

Adriani Nikolakopoulou
Universität Freiburg

Theodoros Papakonstantinou
Universität Freiburg

Georgia Salanti
Universität Bern

Orestis Efthimiou
Universität Bern

Guido Schwarzer
Universität Freiburg

Abstract

Network meta-analysis compares different interventions for the same condition, by combining direct and indirect evidence derived from all eligible studies. Network meta-analysis has been increasingly used by applied scientists and it is a major research topic for methodologists. This article describes the R package **netmeta**, which adopts frequentist methods to fit network meta-analysis models. We provide a roadmap to perform network meta-analysis, along with an overview of the main functions of the package. We present three worked examples considering different types of outcomes and different data formats to facilitate researchers aiming to conduct network meta-analysis with **netmeta**.

Keywords: network meta-analysis, R, inconsistency, component network meta-analysis.

1. Introduction

Network meta-analysis (NMA), also known as multiple-treatments meta-analysis or mixed-treatment comparison, is an extension of pairwise meta-analysis, pooling information from all randomized trials among a set of different interventions for the same medical condition (Salanti 2012). It takes into account and synthesizes both direct and indirect evidence in a single analysis: An estimate of the difference in effects between two given treatments may, in fact, derive from studies directly comparing them (direct evidence), but also from studies included

in paths connecting the two treatments via one or more intermediate comparators (indirect evidence). As in pairwise meta-analysis, studies included in an NMA are assumed to be independent. NMA additionally requires that the transitivity assumption holds. Transitivity suggests that the underlying true relative treatment effect of each comparison is the same across all, observed or not, comparisons. Examination of the similarity of the distribution of effect modifiers across comparisons is often made to judge upon the plausibility of transitivity in a network. Transitivity implies consistency, which means that direct and indirect evidence are in agreement (Salanti 2012).

An NMA provides network estimates for the relative effects between any pair of interventions included in the network, and a ranking thereof, with respect to the outcome of interest. The latter is not possible when performing a series of pairwise meta-analyses, i.e., one per treatment comparison. Given the advantages of NMA and the ease to fit models using common statistical software, its use in applied projects has been increasing rapidly in the last decade (Zarin *et al.* 2017; Petropoulou *et al.* 2017) and it represents a major research topic for methodologists (Efthimiou *et al.* 2015).

R package **netmeta** (Rücker *et al.* 2022) is the only specialized R package for frequentist network meta-analysis. The package adopts the approach by Rücker (2012) and builds on **meta** (Schwarzer 2007; Schwarzer *et al.* 2015; Balduzzi *et al.* 2019), a general R package for pairwise meta-analysis. Frequentist NMA can also be fitted using R packages **metafor** (Viechtbauer 2010) and **mixmeta** (Sera *et al.* 2019), which provide functions for multivariate and multilevel meta-analysis. In fact, **netmeta** internally calls `rma.mv()` from **metafor** to calculate the maximum likelihood and the restricted maximum likelihood estimators for the between-study variance. R package **mixmeta** implements various meta-analytical models, both standard and non-standard, through a unified mixed-effects framework.

Bayesian methods for NMA play an important role both in methodological research and applications (Lu and Ades 2004; Salanti *et al.* 2008; Welton *et al.* 2009). This is reflected in the large number of R packages for Bayesian NMA: **gemtc** (van Valkenhoef and Kuiper 2021), **bnma** (Seo and Schmid 2022), **pcnetmeta** (Lin *et al.* 2020), **multinma** (Phillippo 2022), **nmaINLA** (Guenhan *et al.* 2018), **bayesmeta** (Röver 2020), **BUGSnet** (Béliveau *et al.* 2019).

Package **netmeta** (Rücker *et al.* 2022) is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=netmeta`. According to **cranlogs** (Csárdi 2019), R package **netmeta** is the most popular R package for network meta-analysis (3 645 monthly downloads between November 2021 and October 2022) followed by **gemtc** (1 781 downloads) and **bayesmeta** (919 downloads). Main reasons for its popularity are presumably the large number of implemented statistical methods and the easy application of these methods. More details on these aspects are provided in this article which describes R package **netmeta** in detail.

The structure of the paper is as follows. Three example data sets are introduced in Section 2: Models implemented in **netmeta**, i.e., the standard NMA model (Rücker 2012; Rücker and Schwarzer 2014), methods specific for rare binary outcomes (Efthimiou *et al.* 2019), and component NMA (Welton *et al.* 2009; Rücker *et al.* 2020b), are described in Section 3. A roadmap to perform an NMA, along with an overview of the main functions of **netmeta** is provided in Section 4. The three worked examples are presented in Section 5.

2. Data sets

We will use three data sets which are available in R package **netmeta**. All three examples come from a database of NMAs (Petropoulou *et al.* 2017) made available by R package **nmadb** (Papakonstantinou 2019) which includes a set of functions for accessing the data sets and main characteristics. We extracted the data from **nmadb** and saved them in common data formats for NMA: one row per study, called wide arm-based format and one row per treatment-arm per study, called long arm-based format. The first two data sets consider continuous and binary outcomes and are available in wide and long arm-based format. The third data set, available in long arm-based format, was chosen as an example of a component NMA. Arm-based in either wide or long format means that data on each treatment arm is available, as opposed to contrast-based format, where data are given for each contrast of treatments.

2.1. Data set 1: Adjuvant treatments in Parkinson's disease

The first data set (record no. 480851 in **nmadb**) contains data from a Cochrane review assessing efficacy and safety of three drug classes as adjuvant treatment to levodopa therapy in patients with Parkinson's disease and motor complications (Stowe *et al.* 2010). The authors conducted three pairwise meta-analyses comparing dopamine agonists, catechol-O-methyl transferase inhibitors (COMTI), and monoamine oxidase type B inhibitors (MAOBI) with placebo. The primary outcome was the mean reduction of the time spent in a relatively immobile "off" phase (mean off-time), calculated in hours per day. Relative treatment effects were expressed as mean difference. Data on this outcome were available for 5 331 patients from 28 studies comparing an active treatment with placebo and one three-arm study comparing two active treatments with placebo.

The full data set — in wide arm-based format — is part of R package **netmeta** and made available using the following command.

```
R> data("Stowe2010", package = "netmeta")
```

The data set `Stowe2010` has 29 rows, each containing information regarding a single study. To have a look at the format of this data set, we display some rows including the three-arm study.[1]

```
R> library("dplyr")
R> Stowe2010 %>% slice(18:20)
```

```
              study id    t1    y1  sd1  n1      t2    y2  sd2  n2
1   COMTI(E) INT-OZ 18 COMTI -0.86 2.90  99 Placebo -0.54 3.13  63
2            LARGO 19 COMTI -1.20 2.21 218 Placebo -0.40 2.21 218
3 COMTI(E) Nomecomt 20 COMTI -1.30 2.20  85 Placebo -0.10 2.45  86
     t3    y3  sd3  n3
1          NA   NA  NA
2 MAOBI -1.18 2.23 222
3          NA   NA  NA
```

[1] We make R package **dplyr** (Wickham *et al.* 2022) solely available for data processing and manipulation. Network meta-analysis commands in **netmeta** do not depend on **dplyr**.

All studies with a single pairwise comparison provide information for two means (`y1`, `y2`), two standard deviations (`sd1`, `sd2`), two group sample sizes (`n1`, `n2`), along with two treatment labels (`t1`, `t2`). Information for a third treatment (`t3`, `y3`, `sd3`, `n3`) is not available for two-arm studies and thus coded as missing. The three-arm LARGO study provides information for all variables. A disadvantage of this wide arm-based format is that we would have to add four more variables (`t4`, `y4`, `sd4`, `n4`) for a four-arm study.

2.2. Data set 2: Antithrombotics to prevent strokes

The second data set (record no. 501250 in **nmadb**) comes from a systematic review aiming to estimate the effects of eight antithrombotic treatments in reducing the incidence of major thrombotic events in patients with non-valvular atrial fibrillation (Dogliotti *et al.* 2014). The review included 20 studies (79 808 participants), four of which were three-arm studies. The primary outcome is stroke reduction.

The full data set — in long arm-based format — is also part of R package **netmeta**.

```
R> data("Dogliotti2014", package = "netmeta")
R> Dogliotti2014 %>% head(7)
```

```
          study id       treatment stroke total
1 AFASAK-I 1989  1            VKAs      9   335
2 AFASAK-I 1989  1         Aspirin     16   336
3 AFASAK-I 1989  1 Placebo/Control     19   336
4   BAATAF 1990  2            VKAs      3   212
5   BAATAF 1990  2 Placebo/Control     13   208
6     CAFA 1991  3            VKAs      6   187
7     CAFA 1991  3 Placebo/Control      9   191
```

The first three rows pertain to the three-arm AFASAK-I study. For each treatment arm, the treatment label (`treatment`), the number of strokes (`stroke`), and the group sample size (`total`) is available. The subsequent rows belong to the two-arm BAATAF and CAFA studies. An advantage of this long arm-based format is that a four-arm study could be easily added by four additional rows.

2.3. Data set 3: Treatments for chronic obstructive pulmonary disease

The third data set (record no. 501201 in **nmadb**) comes from a systematic review of randomized controlled trials on pharmacologic treatments for chronic obstructive pulmonary disease (COPD, Baker *et al.* 2009). The primary outcome, occurrence of one or more episodes of COPD exacerbation, is binary (yes/no). For this outcome, five drug treatments (fluticasone, budesonide, salmeterol, formoterol, tiotropium) and two combinations (fluticasone + salmeterol, budesonide + formoterol) were compared to placebo. The authors considered the two combinations as separate treatments instead of evaluating the individual components which is not surprising as the first methodological publication on how to evaluate individual components in NMA was published in the same year (Welton *et al.* 2009).

In this NMA, 38 studies (29 two-arm, three three-arm, six four-arm studies with a total of 28 216 patients) were included.

```
R> data("Baker2009", package = "netmeta")
R> subset(Baker2009, study == "Mahler 2002")
```

```
         study year id              treatment exac total
28 Mahler 2002 2002 14            Fluticasone   77   168
29 Mahler 2002 2002 14             Salmeterol   63   160
30 Mahler 2002 2002 14 Fluticasone+Salmeterol   68   165
31 Mahler 2002 2002 14                Placebo   79   181
```

The data set is again in long arm-based format. For each treatment arm, the treatment label (`treatment`), the number of patients with COPD exacerbations (`exac`), and the group sample size (`total`) is available. We can see that the Mahler study has four arms, comparing fluticasone, salmeterol, and their combination to placebo.

3. Network meta-analysis in theory

The notation in this paper follows Schwarzer *et al.* (2015, Chapter 8). Let n be the number of treatments of interest in a network (also called nodes or vertices). Each study contributes a number of pairwise comparisons, for example three-arm studies contribute three pairwise comparisons and four-arm studies contribute six pairwise comparisons. The sum of all pairwise comparisons across studies is denoted as m. Let k be the number of independent studies. If there are only two-arm studies, m corresponds to k, while m is greater than k if at least one study evaluates more than two treatments.

3.1. Standard NMA model

The common (fixed) effects and random effects NMA model developed by Rücker (2012) is based on graph and electrical network theory (Bailey 2007). In order to fit this model, treatment estimates of all pairwise comparisons $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \ldots, \hat{\theta}_m)^\top$ and corresponding standard errors $s_i, i = 1, \ldots, m$, must be available. A multi-arm study with p treatments will contribute $p(p-1)/2$ comparisons, meaning that treatment effects and standard errors need to be provided for each of them, e.g., a three-arm study will contribute three pairwise comparisons and a four-arm study will contribute six pairwise comparisons. As is common in meta-analysis, standard errors are assumed to be known and fixed. For multi-arm studies, adjusted standard errors are utilized (Rücker and Schwarzer 2014). In addition, a $m \times n$ design matrix $\mathbf{X}$ is required for the analysis which is defined by the network geometry, i.e., the way that m pairwise comparisons connect the n treatments. The elements of $\mathbf{X}$ are 0, 1 or -1 and represent the treatments (columns) of the comparison in the respective row.

The first step to get network treatment estimates $\hat{\boldsymbol{\theta}}^{\text{NMA}}$ is the calculation of the Moore-Penrose pseudoinverse matrix (Albert 1972). To this aim, we define the $n \times n$ Laplacian matrix $\mathbf{L}$ as

$$\mathbf{L} = \mathbf{X}^\top \mathbf{W} \mathbf{X} \tag{1}$$

where $\mathbf{W}$ is a diagonal matrix of dimension $m \times m$ whose diagonal elements are the inverse variance weights $(1/s_1^2, \ldots, 1/s_m^2)$. Matrix $\mathbf{L}$ has rank $n-1$ and is thus singular. However, its Moore-Penrose pseudoinverse $\mathbf{L}^+$ (Rao and Mitra 1971; Gutman and Xiao 2004) is defined

and can be calculated by

$$\mathbf{L}^{+} = (\mathbf{L} - \mathbf{J}/n)^{-1} + \mathbf{J}/n$$

where $\mathbf{J}$ is a $n \times n$ matrix whose elements are all 1.

Once we have $\mathbf{L}^{+}$, we can finally calculate the network treatment estimates $\hat{\boldsymbol{\theta}}^{\text{NMA}}$ as

$$\hat{\boldsymbol{\theta}}^{\text{NMA}} = \mathbf{X}\mathbf{L}^{+}\mathbf{X}^{\top}\mathbf{W}\hat{\boldsymbol{\theta}} = \mathbf{H}\hat{\boldsymbol{\theta}} \tag{2}$$

where $\mathbf{H}$ is typically called the hat matrix. From (2) we can see that elements of $\hat{\boldsymbol{\theta}}^{\text{NMA}}$ are linear combinations of elements from $\hat{\boldsymbol{\theta}}$ (the observed estimates), with coefficients coming from rows of $\mathbf{H}$.

The variance-covariance matrix of $\hat{\boldsymbol{\theta}}^{\text{NMA}}$ can be calculated as $\mathbf{X}\mathbf{L}^{+}\mathbf{X}^{\top}$ which can be used to estimate all treatment contrasts and associated standard errors. These estimates are the same as obtained by weighted maximum likelihood (Yates 1940; Paterson 1983; Senn *et al.* 2013).

A random effects model can be defined assuming a common heterogeneity variance τ^2 for each pairwise treatment comparison. To fit a random effects model, an estimate of τ^2 is added to the variance of each comparison, $s_i^2 + \hat{\tau}^2, i = 1, \ldots, m$ before calculating $\mathbf{L}$ in (1). A special case of the generalised DerSimonian-Laird estimate is a common estimator for τ^2 (Jackson *et al.* 2013; Rücker and Schwarzer 2014). Maximum likelihood and restricted maximum likelihood estimates for τ^2 are also available (Jackson *et al.* 2011; White 2015).

3.2. Ranking of treatments

Several ranking metrics exist to summarize the results from NMA and produce a treatment hierarchy. One can calculate the probability of each treatment being at each possible rank. This is traditionally done in a Bayesian framework drawing from the posterior distribution of network treatment estimates, but is also possible in a frequentist environment using resampling techniques. Surface under the cumulative ranking curve (SUCRA) values provide a summary of the rankograms (Salanti *et al.* 2011). The name originates from the fact that SUCRA is calculated as the surface under the cumulative ranking probability curve. P-scores are a frequentist analogue to SUCRA values without resampling methods (Rücker and Schwarzer 2015). SUCRA and P-score express the extent of certainty that a treatment is better than another treatment, averaged over all competing treatments.

3.3. Evaluating heterogeneity and inconsistency

The overall heterogeneity / inconsistency statistic Q_{total} is defined as

$$Q_{\text{total}} = (\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}^{\text{NMA}})^{\top}\mathbf{W}(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}^{\text{NMA}}).$$

This statistic measures the total heterogeneity / inconsistency in the network. When all studies are two-arm studies, Q_{total} follows a χ^2 distribution with $k - (n-1)$ degrees of freedom under the null hypothesis of no heterogeneity, conditional on the (assumed to be known) comparison-specific variances. More general, each p-arm study contributes $p - 1$ degrees of freedom to Q_{total}. The sum of the degrees of freedom contributed by each study minus $n - 1$ (the number of treatments minus 1, which is the dimension of the consistent subspace) gives the total degrees of freedom denoted by df in the following.

A generalised I^2 statistic (Higgins and Thompson 2002) can be defined as

$$I^2 = \max\left(\frac{Q_{\text{total}} - \text{df}}{Q_{\text{total}}}, 0\right).$$

Defining a design as a unique set of treatments compared in an individual study (Higgins *et al.* 2012), the maximum number of designs is $2^n - n - 1$ for n treatments. This leads to four possible designs for three treatments A, B, and C: A:B, A:C, B:C, A:B:C; or 11 designs for four treatments. Usually not all possible designs are present in an NMA. In a pairwise meta-analysis, all trials have the same design, e.g., A:B.

In some cases, inconsistency between designs might be likely, when they differ systematically: for example, when all two-arm studies in a network are older, while three-arm studies are more recent, design is a potential source of inconsistency.

Through an appropriate decomposition of the Q_{total} statistic (Krahn *et al.* 2013)

$$Q_{\text{total}} = Q_W + Q_B$$

we can determine the heterogeneity of study results within the same design, Q_W, and the inconsistency in treatment effects between different designs, Q_B. A decomposition of Q_W into parts coming from each study and a decomposition of Q_B into parts coming from each design is also possible. Thus, for assessing the global inconsistency in a random effects model, the Q_B statistic can be calculated based on a full design-by-treatment interaction random effects model (Higgins *et al.* 2012), considering the "method of moments" estimate for τ^2 (Jackson *et al.* 2012).

A method to evaluate local inconsistency in each treatment comparison separately is by using the separate indirect from direct evidence (SIDE) method (Dias *et al.* 2010). According to SIDE, each network estimate is split into the direct and the indirect estimate which are then tested against each other. A z test of the difference between direct and indirect estimate indicates potential evidence for inconsistency for each comparison in the network. A similar method, originally implemented in a Bayesian framework is back-calculation (Dias *et al.* 2010). Using this method, an indirect estimate with its variance is derived from the respective quantities derived by pairwise and network meta-analysis. Similarly to SIDE, the difference between direct and indirect estimate is used to construct a test statistic to estimate inconsistency.

3.4. Specific NMA methods for rare binary outcomes

When the outcome of interest is binary and studies provide number of events and sample sizes in each treatment group as in Dogliotti *et al.* (2014), we can estimate relative treatment effects (e.g., log odds ratios) and corresponding standard errors from each study. Subsequently, we can perform a standard NMA as described in Section 3.1.

The estimation of study-specific odds or risk ratios is based on approximations that do not perform well when events are rare either because event rates are low or group sample sizes are small. Especially treatment estimates can assume infinite values in studies with one or more zero event numbers. This can be circumvented via a "continuity correction"; this, however, may lead to low model performance. Conversely, there are two NMA models explicitly designed for rare binary outcomes currently supported by **netmeta**, the non-central hypergeometric (NCH-NMA) and the Mantel-Haenszel (MH-NMA) model.

The NCH-NMA model was proposed by Stijnen *et al.* (2010). It models the likelihood of events in each study arm conditional on the total number of events in the study via a noncentral hypergeometric distribution. In **netmeta** this is done using the so-called Breslow approximation, which is only valid when the numbers of events are small relative to the group sizes. Of note, **netmeta** provides only a common effects version of the NCH-NMA model and maximizes the combined log-likelihood of all studies to estimate all log odds ratios in the network, as well as their variance-covariance matrix. Studies with zero events in all treatment arms do not contribute to the likelihood.

The second method, MH-NMA, was proposed by Efthimiou *et al.* (2019) and generalizes the Mantel-Haenszel method for pairwise meta-analysis (Mantel and Haenszel 1959) which is a common effects method. The analysis is conducted in three stages. At stage one, studies with total zero events are removed and studies are grouped afterwards by design. Within each design, treatments with zero events in all studies are removed and designs left with only one treatment are removed.

At stage two, log odds ratios are estimated within each design. For two-arm studies, we employ the usual MH estimator (Mantel and Haenszel 1959). For a design d including p treatments, we need $p - 1$ odds ratios. To calculate them, we define $c_{XYi} = a_{Xi}b_{Yi}$ / $t_{(+i)}$, with i indicating the study index, a_{Xi} and b_{Yi} being the number of events and non-events within treatment arms X and Y respectively and $t_{(+i)}$ being the total sample size of study i. We define as C_{XY} the sum of all c_{XYi} over all studies of the same design and we then define $L_{XY} = \ln(C_{XY}/C_{YX})$. The log odds ratio for XY for design d is then estimated as $\hat{\theta}_{d,XY} = \frac{L_{X+}-L_{Y+}}{p}$ where $L_{X+} = \sum_{J=1}^{p} L_{XJ}$, likewise for other log odds ratios from this design. The expressions for the variance-covariance matrix are provided elsewhere (Greenland 1989; Efthimiou *et al.* 2019). At the end of this stage, we obtain a vector of log odds ratios, and corresponding variance-covariance matrix for design d.

Finally, at the third stage we synthesize the design-specific estimates across the network assuming consistency. All treatment contrasts versus an arbitrarily chosen reference treatment are the basic parameters of the model. The network treatment effects for all comparisons in the network and their variance-covariance matrix are then estimated using weighted least squares regression.

3.5. Component NMA

Treatments in NMA may sometimes be combinations of several interventions, i.e., may be composite sharing one or more common components, see Baker *et al.* (2009) for an example. The standard approach in analyzing such data sets is an NMA where all existing (simple or composite) treatments in the network are considered as different nodes in the network. Methods for such a standard analysis are described in Section 3.1.

However, we might be interested in estimating the individual effect of each component, to help us identify the best combination of components. If some treatments are combinations of common components, an additive component NMA (CNMA) model can be used to evaluate the effect of each component (Welton *et al.* 2009). This model assumes that the effect of a treatment combination is the sum of the effects of its components, which implies that common components cancel out in comparisons, and do not impact on the relative treatment effects.

While Welton *et al.* (2009) describe Bayesian methods for CNMA, R package **netmeta** implements these models under a frequentist framework (Rücker *et al.* 2020b, 2021).

The design matrix of the additive CNMA model is the $n \times c$ matrix

$$\mathbf{X} = \mathbf{BC}. \tag{3}$$

Matrix $\mathbf{X}$ combines information on the structure of the network ($\mathbf{B}$ with m rows, representing the pairwise comparisons, and n columns, representing the treatments) and information on how the n treatments are composed of the c components ($n \times c$ matrix $\mathbf{C}$). The elements of $\mathbf{X}$ are 0, 1 or -1 and represent the components (columns) of the comparison in each row. The additive model (common effects version) is given by

$$\hat{\theta} = \mathbf{X}\beta + \epsilon, \epsilon \sim \mathcal{N}(0, \mathbf{\Sigma})$$

where $\hat{\theta}$ is the vector of observed relative effects from the studies, $\mathbf{X}$ the design matrix given in (3), and β a parameter vector of length c, representing the components, which is estimated using weighted least squares regression. For details of the estimation see Rücker *et al.* (2020b). Interaction CNMA models are readily available by additional columns in the components matrix $\mathbf{C}$, thus allowing for synergistic or antagonistic effects when combining two components. In some cases CNMA models can also be applied to networks that are disconnected (i.e., they consist of two or more separate subnetworks).

4. Network meta-analysis in practice – Overview of netmeta

R package **netmeta** implements the (C)NMA approaches introduced in the previous section. In this section we describe a workflow to perform a (C)NMA, along with the related functions of **netmeta** (Figure 1). After installing **netmeta**, we make it available for the current R session.

```
R> library("netmeta")

Loading required package: meta

Loading 'meta' package (version 6.1-0).
Type 'help(meta)' for a brief overview.
Readers of 'Meta-Analysis with R (Use R!)' should install
older version of 'meta' package: https://tinyurl.com/dt4y5drs

Loading 'netmeta' package (version 2.7-0).
Type 'help("netmeta-package")' for a brief overview.
Readers of 'Meta-Analysis with R (Use R!)' should install
older version of 'netmeta' package: https://tinyurl.com/kyz6wjbb
```

We can see from the printout that R package **meta** (Balduzzi *et al.* 2019) is also loaded, which provides some basic functionality, and that brief overviews of methods and general hints on **meta** and **netmeta** are available in the R help system.

The following command specifies that treatment estimates and confidence interval limits should be printed with two digits and standard errors with three digits.

```
R> settings.meta(digits = 2, digits.se = 3)
```

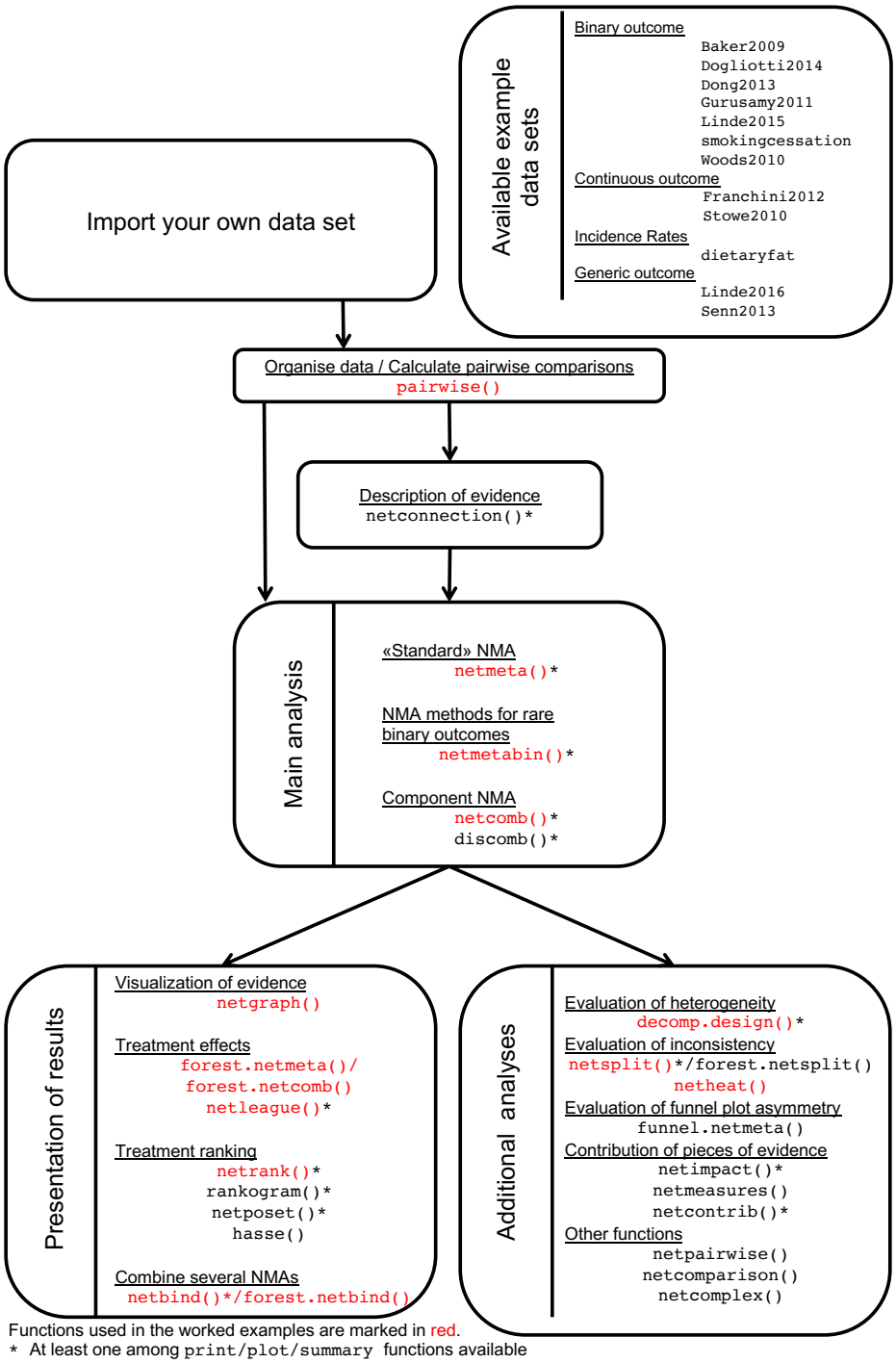


Figure 1: Workflow to perform (component) network meta-analysis with R package **netmeta**.

These settings are recognized by all print and plot functions for (network) meta-analysis objects in the current R session.

4.1. Organise data and perform main analysis

The main function of **netmeta** for standard NMA is likewise called `netmeta()`. It needs the data to be in contrast-based format, which means that each row of the data set contains information from one pairwise comparison. For multi-arm studies, all pairwise comparisons must be provided, e.g., a four-arm study must contribute six pairwise comparisons (see Section 3.1). Furthermore, `netmeta()` expects that the estimated treatment effects (argument `TE`) and corresponding standard errors (argument `seTE`) are available for all pairwise comparisons as well as information on the two treatment groups (arguments `treat1` and `treat2`). Finally, `netmeta()` must know which rows belong to each multi-arm study, as standard errors are adjusted to take the correlation between pairwise comparisons into account (Rücker and Schwarzer 2014). Accordingly, argument `studlab` is mandatory for a network with at least one multi-arm study and study labels must be identical. For example, "LARGO", "Largo", and "Largo" for the pairwise comparisons of the three-arm study in `Stowe2010` would be considered as three independent studies.

Typically, a data set is not in the contrast-based format expected by `netmeta()` and must be pre-processed accordingly. Auxiliary function `pairwise()` can be used to transform a data set from wide or long arm-based format to the contrast-based format and to calculate all pairwise treatment comparisons. The function can be used with binary, continuous, generic outcomes or incidence rates by internally calling `metabin()`, `metacont()`, `metagen()`, or `metainc()` from R package **meta** to calculate the treatment estimates and standard errors. An R object created with `pairwise()` contains all information to conduct an NMA and can be used as single input to `netmeta()`.

R function `netmeta()` has several additional arguments, some of which are described below. By default, both common and random effects NMA are conducted. This can be changed using the logical arguments `common` and `random`. The method to estimate the between-study variance can be specified by `method.tau`. Finally, a reference treatment can be specified with argument `reference.group`; this only affects the way results are presented, but not the analysis itself. The first treatment in the network is used as reference if argument `reference.group` was not specified in `pairwise()` or `netmeta()`.

The function `netconnection()` can be used to obtain information on the network structure and to determine whether a given network is fully connected or consists of subnetworks. Again, an R object created with `pairwise()` can be used as single input to `netconnection()`. For disconnected networks, use of `netmeta()` will result in an error with the recommendation to use `netconnection()` for further information.

R functions `pairwise()` and `netconnection()` can also be used in connection with rare binary outcomes or when conducting a component NMA. Likewise, many methods described in the next two sections are also relevant to rare binary outcomes and CNMAs.

4.2. Presentation of network structure, results and rankings

In general, the first step of an NMA is to construct a network graph in order to get an overview of the network structure. After conducting the NMA, results can be summarized using forest plots or league tables. A ranking of all treatments evaluated in an NMA provides additional information on the merits of individual treatments.

Network graphs

A graphical presentation of the network can be obtained through `netgraph()` which can be used with an R object created with `netconnection()` or `netmeta()`. Each treatment is represented by a point (node) in the plane and treatments are connected by a line (edge), if at least one direct pairwise comparison exists. By default, multi-arm studies are not highlighted in a network graph. If argument `multiarm` is `TRUE`, three-arm studies are indicated by a different color for the area enclosing the respective nodes. Studies with more than three treatments cannot be visually marked in the network graph. The standard layout is a circle presentation with equally distanced treatments on the circle. Using argument `seq = "optimal"` (which can be abbreviated), the number of line crossings is aimed to be minimized. Other layouts are available with `iterate = TRUE` (Rücker and Schwarzer 2016) which optimize the distance between treatment nodes using a stress majorization algorithm (Kamada and Kawai 1989; Hu 2012).

R function `netgraph()` has several additional arguments; three of them, more commonly used, are the following: edges of the network have a 3-D look if argument `plastic = TRUE` (default) and the number of studies contributing to each pairwise comparison can be printed on the edges (`number.of.studies = TRUE`). Finally, the width of the edges is determined by argument `thickness`. By default, edges are proportional to the number of studies directly comparing treatments.

Forest plots

Forest plots can be created using the generic function `forest()` or `plot()` which can be used, amongst others, with '`netmeta`' objects. Basic settings include specifying whether results of the common or random effects model should be shown, and which treatment(s) should be used as reference (argument `reference.group`). In general, the settings from '`netmeta`' objects are used as defaults. Results of the random effects model are shown in the forest plot if both a common and random effects NMA were conducted (can be changed using argument `pooled`). The first treatment in the network meta-analysis is used as reference if argument `reference.group` was not specified in `netmeta()` and `forest()`.

Other important arguments include `sortvar` to change the order of treatments in the forest plot and the logical arguments `drop.reference.group` and `baseline.reference` to not print a line with the reference group or to switch the order of reference and other treatments, respectively.

League tables

A concise way to present NMA results is a league table (Hutton *et al.* 2015) which is a square matrix showing all pairwise comparisons in an NMA. For a single NMA, network estimates are typically shown in the lower triangle and estimates from direct pairwise comparisons in the upper triangle. It is also common to show results from two NMAs in a single league table, e.g., network estimates from an efficacy outcome in the lower triangle and a safety outcome in the upper triangle.

R function `netleague()` can be used to create league tables for '`netmeta`' objects. Either one or two NMAs can be provided as input. Furthermore, the user can specify whether to construct a league table for common or random effects network estimates (arguments `common`

and `random`), and how to sort the league table (argument `seq`). Finally, league tables can be easily exported, e.g., as Excel sheets, to be further edited for a publication.

Ranking of treatments

R function `rankogram()` can be used to calculate the probabilities of each treatment being at each possible rank, and SUCRA values for a '`netmeta`' object. A pivotal argument is `small.values` determining whether small values in the outcome are beneficial (`"good"`) or harmful (`"bad"`). If argument `small.values` is not provided in the `netrank()` command, it is taken from the '`netmeta`' object which defaults to `"good"`. Counter-intuitive ranking results are typically due to using the wrong value for argument `small.values`. R function `plot.rankogram()` plots these probabilities in a separate graph for each treatment, commonly referred to as rankograms.

R function `netrank()` can be used to generate a treatment hierarchy for a '`netmeta`' or '`rankogram`' object using P-score or SUCRA. The ranking metric to be calculated can be selected using argument `method`. The meaning of the `small.values` argument is the same as in the `rankogram()` function.

Additional functions are available to (partially) rank treatments for more than one outcome (Rücker and Schwarzer 2017). R function `netposet()` creates a partial order of treatment ranks. Input to `netposet()` can be any number of NMAs created with `netmeta()` or an existing ranking matrix. Function `plot.netposet()` can be used to create a scatter plot for rankings of two outcomes and `hasse()` can be used to generated a Hasse diagram (Carlsen and Bruggemann 2014) for any number of outcomes.

4.3. Evaluation of heterogeneity and inconsistency

The decomposition of the overall Q statistic described in Section 3.3 can be performed with `decomp.design()` for '`netmeta`' objects.

The function `netsplit()` provides two methods to compare direct and indirect evidence for each pairwise comparison. The first method (argument `method = "Back-calculation"`), separate indirect from direct evidence (SIDE), was introduced by Dias *et al.* (2010) and a corresponding method using back-calculation was described by König *et al.* (2013). The second method (`"SIDDE"`), separate indirect from direct design evidence (SIDDE) was described in Efthimiou *et al.* (2019). For the random effects model, the direct treatment estimates are based on the common between-study variance τ^2 from the NMA. By default, SIDE is used for '`netmeta`' objects and SIDDE for '`netmetabin`' objects. The output for SIDE or SIDDE can be very extensive for large networks. Accordingly, argument `show = "both"` can be used in `print.netsplit()` and `forest.netsplit()` to only show pairwise comparisons contributing both direct and indirect evidence.

The net heat plot (Krahn *et al.* 2013) implemented in `netheat()` visualizes hot spots of inconsistency in an NMA. Note that the net heat plot is a visualization of the "between-designs" part of function `decomp.design()`. The rows and columns of the plot correspond to pairwise comparisons within designs. Not all comparisons are given in the net heat plot, only those that can contribute to inconsistency. The colors on the diagonal represent the inconsistency contribution of the corresponding design (red = large). The colors on the off-diagonal are associated with the change in inconsistency between direct and indirect evidence in a network estimate in the row after relaxing the consistency assumption for the effect of

one design in the column. A blue-colored element indicates that the evidence of the design in the column supports the evidence in the row, whereas a red-colored element indicates that the evidence of the design in the column contrasts to the evidence in the row (Krahn *et al.* 2013).

4.4. Additional analyses

The function `netbind()` can combine different (C)NMA objects. This is useful when the aim is to display the results of several (C)NMA in a forest plot.

The function `funnel.netmeta()` generates a "comparison-adjusted" funnel plot to assess funnel plot asymmetry in an NMA (Chaimani and Salanti 2012; Chaimani *et al.* 2013). The user must specify a meaningful treatment order (argument `order`) which should reflect the (a priori) expected direction of small study effects in each pairwise comparison. See Chaimani and Salanti (2012) for more details on specifying the order.

The function `netimpact()` evaluates the importance of individual studies in an NMA by calculating the reduction of the precision when the study is removed from the network (Rücker *et al.* 2020a). The function `netcontrib()` gives the percentage contribution of each direct comparison (or study) to each network estimate using an algorithm based on flow decomposition (Papakonstantinou *et al.* 2018) or a random walk approach (Davies *et al.* 2022).

The function `netmeasures()` provides measures for quantifying the direct evidence proportion (i.e., the contribution of direct effect estimates combined for two-arm and multi-arm studies to each network estimate), the mean path length, and other measures (König *et al.* 2013).

Finally, `netpairwise()` conducts separate pairwise meta-analyses for all comparisons with direct evidence. In contrast to `netmeta()` and `netsplit()` unadjusted standard errors are used in the calculations and the between-study heterogeneity variance is allowed to differ between comparisons.

4.5. Specific NMA methods for rare binary outcomes: `netmetabin()`

The `netmetabin()` function can be used to perform an NMA for rare binary outcomes, as described in Section 3.4. Like `netmeta()`, the function requires data to be supplied in contrast-based format, where each row contains the number of events and sample sizes from the respective pairwise comparison. This data format can be obtained using `pairwise()`. Three NMA methods for binary outcomes are available: NCH-NMA (`method = "NCH"`), MH-NMA (`"MH"`, default), and the standard NMA model (`"Inverse"`) by Rücker (2012). For NCH-NMA and MH-NMA, only the odds ratio can be used as effect measure (argument `sm`). For the inverse variance method, all effect measures provided by `metabin()` from **meta** can be used, however, this method is typically not suitable for rare binary outcomes. Of note, `pairwise()` and `netmeta()` are called internally for the inverse variance method.

By default, `netmetabin()` excludes studies with zero events in all arms and does not use a continuity correction for studies with zeros in some, but not all treatment arms. This may lead to designs having only one treatment left. These designs are excluded from the network meta-analysis, which may result in network connectivity issues, i.e., the network may become disconnected. A continuity correction can be used in such cases with argument `cc.pooled = TRUE` which uses a continuity correction by adding a small increment (as defined by argument `incr`) to all events and non-events in a study with some zero events. However, as noted in

Section 3.4, using a continuity correction may lead to low model performance.

4.6. Component NMA: `netcomb()`, `discomb()`

CNMA models, described in Section 3.5, are implemented in `netcomb()`. The main argument of `netcomb()` is an R object created with `netmeta()`. A common separator must be used for all combined treatments in the network and by default the plus sign is used in `netcomb()` (argument `sep.comps = "+"`). Accordingly, for combinations named in the form $A + B$ (with treatment components A and B), `netcomb()` automatically conducts a CNMA and estimates the effects of all treatment components and combined treatments in the network, based on the additive model. The separator must be a single character, for example, setting argument `sep.comps = "and"` will result in an informative error message. Finally, specifying a separator that is not available in any treatment label will give the same results as `netmeta()` as each treatment combination will be handled as a separate treatment.

A special property of CNMA is that it is possible to analyze disconnected networks (i.e., networks that consist of two or more separate subnetworks) if subnetworks share at least one common treatment component. For example, if one subnetwork contains treatment $A + B$ and another subnetwork contains treatment $A+C$, then A is a common component of the two networks. A CNMA analysis of a disconnected network can be conducted using `discomb()`. As standard NMA cannot be applied to a disconnected network, `discomb()` needs to start from scratch instead of using a network meta-analysis object. Accordingly, the function shares many arguments with `netmeta()` and `netcomb()` and can be used with a '`pairwise`' object as single input (Rücker *et al.* 2021).

Functions `netcomplex()` and `netcomparison()` can be used to calculate the treatment effect for any complex intervention, i.e., combination of treatment components, or comparisons of complex interventions. Input is an object created with `netcomb()` or `discomb()`.

5. Three worked examples

5.1. Example 1: Adjuvant treatments in Parkinson's disease

This data set is in wide arm-based format and includes several two-arm studies and a single three-arm study. Accordingly, we can use `pairwise()` to transform the data set into contrast-based format and to estimate treatment effects and corresponding standard errors.

```
R> pw1 <- pairwise(treat = list(t1, t2, t3), n = list(n1, n2, n3),
+    mean = list(y1, y2, y3), sd = list(sd1, sd2, sd3),
+    studlab = study, data = Stowe2010)
```

For a continuous outcome, we have to specify all variables containing information on group sample sizes (argument `n`), means (`mean`), and standard deviations (`sd`). Furthermore, we have to provide the treatment labels (`treat`) and—as we have multi-arm studies—the study labels (`studlab`). For a data set in wide arm-based format, the information for these quantities with exception of the study labels is distributed over several variables, e.g., `t1`, `t2`, and `t3` for treatment labels. Therefore, we are using `list()` to collate all respective variables. Finally, we specify the data set in argument `data`. By default, the mean difference is used as

effect measure for continuous outcomes (argument `sm = "MD"`). Alternatively, we could use standardized mean differences (`"SMD"`) or ratio of means (`"ROM"`). For the standardized mean difference, the method by Crippa and Orsini (2016) is used to guarantee consistent SMDs and standard errors for multi-arm studies.

R object `pw1` is a data frame with 31 rows, each corresponding to a pairwise comparison (28 rows from 28 two-arm studies and 3 rows from the three-arm study), and 25 columns (variables).

```
R> dim(pw1)

[1] 31 25

R> names(pw1)

 [1] "TE"       "seTE"     "studlab"  "treat1"   "treat2"   "n1"
 [7] "mean1"    "sd1"      "n2"       "mean2"    "sd2"      "study"
[13] "id"       "t1"       "y1"       "sd1.orig" "n1.orig"  "t2"
[19] "y2"       "sd2.orig" "n2.orig"  "t3"       "y3"       "sd3"
[25] "n3"
```

The first 11 variables `TE` to `sd2` have been created by `pairwise()` and will be described in more detail below. The other variables come directly from `Stowe2010` and some variables have been renamed, e.g., `sd1.orig` and `n1.orig`, as the original names coincide with automatic generated variables.

The following commands show all rows in `pw1` coming from the three studies we already considered in Section 2.1. We see that the two-arm studies contribute a single row whereas the three-arm study contributes three rows.

```
R> selstudy1 <- c("COMTI(E) INT-OZ", "LARGO", "COMTI(E) Nomecomt")
R> subset(pw1, studlab %in% selstudy1) %>%
+    select(TE, seTE, studlab, treat1, treat2, mean1, mean2)

      TE      seTE           studlab  treat1  treat2 mean1 mean2
18 -0.32 0.4903630   COMTI(E) INT-OZ   COMTI Placebo -0.86 -0.54
19 -0.80 0.2116796             LARGO   COMTI Placebo -1.20 -0.40
20 -0.02 0.2116709             LARGO   COMTI   MAOBI -1.20 -1.18
21  0.78 0.2116709             LARGO Placebo   MAOBI -0.40 -1.18
22 -1.20 0.3560024 COMTI(E) Nomecomt   COMTI Placebo -1.30 -0.10
```

The first two variables contain the mean differences (`TE`) and corresponding standard errors (`seTE`) of the pairwise comparison defined by variables `treat1` and `treat2`. Mean differences are calculated as differences between variables `mean1` and `mean2`.

The first five variables in object `pw1` are the main input to `netmeta()`.

```
R> net1 <- netmeta(TE, seTE, treat1, treat2, studlab, data = pw1,
+    common = FALSE, ref = "plac")
```

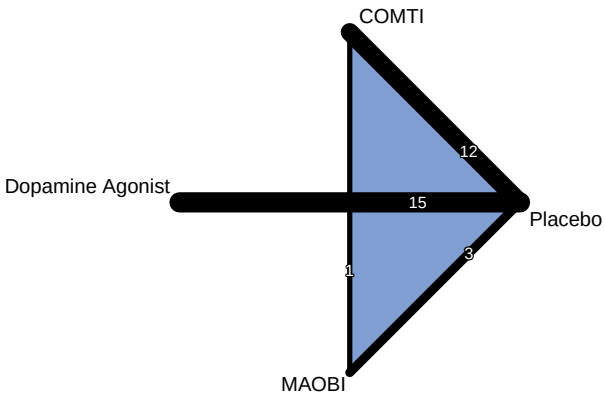


Figure 2: Network graph for Parkinson's disease network meta-analysis (Stowe *et al.* 2010). Line widths are proportional to the number of studies directly comparing treatments.

Here, we specify that we want to only see results for the random effects model (`common = FALSE`) and use placebo as reference (`reference.group`); otherwise the first treatment would be used as reference.

We can also use object `pw1` directly to conduct the NMA.

```
R> net1 <- netmeta(pw1, common = FALSE, ref = "plac")
```

Network graphs

Before printing the results of the NMA, we have a look at the network graph, which does not use the 3-D look (argument `plastic`), marks the three-arm study (`multiarm`) and includes the number of studies for each pairwise comparison (`number.of.studies`). The other arguments determine the size and position of treatment labels (`cex`, `offset`) and the size and position of the number of studies on the lines connecting treatments (`cex.number`, `pos.number.of.studies`). Here, the use of argument `pos.number.of.studies` is important as the two numbers for the comparisons dopamine agonist versus placebo and COMTI versus MAOBI would otherwise lie on top of each other in the middle of the network graph.

```
R> netgraph(net1, plastic = FALSE, multiarm = TRUE,
+    cex = 1.25, offset = ifelse(trts != "MAOBI", 0.025, 0.01),
+    number.of.studies = TRUE, cex.number = 1, pos.number.of.studies = 0.3)
```

The network graph is shown in Figure 2. The line width is proportional to the number of studies directly comparing the treatments and the three-arm study is marked by the blue area. We immediately see that the comparisons dopamine agonist versus placebo and COMTI versus placebo include the largest number of studies. Furthermore, the comparison COMTI versus MAOBI was only considered in the three-arm study.

We may want to avoid the crossing lines in this small network which is due to the circle presentation. We can switch to a different layout using argument `iterate = TRUE`. A disadvantage of all layouts but the circle presentation is that treatment labels may overlap with edges. In the network graph generated with the following command (figure not shown) the treatment label for placebo is overlapping with an edge.

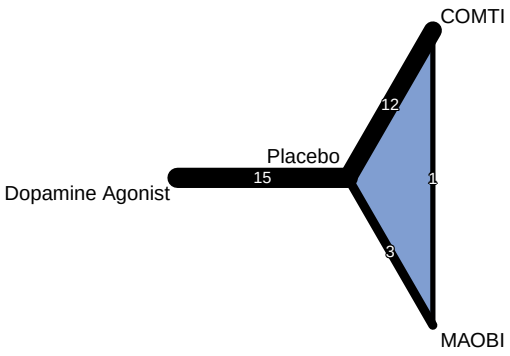


Figure 3: Network graph without crossings for Parkinson's disease network meta-analysis (Stowe *et al.* 2010). Line widths are proportional to the number of studies directly comparing treatments.

```
R> nodes1 <- netgraph(net1, iterate = TRUE)$nodes
```

Each call of R function `netgraph()` creates an invisible list containing data frames with information on nodes and edges. The above command extracts the information for nodes. The x and y adjustment of treatment labels is stored in the variables `adj.x` and `adj.y`.

```
R> nodes1 %>% select(trts, adj.x, adj.y)
```

```
              trts adj.x adj.y
1            COMTI     0     0
2 Dopamine Agonist     1     1
3            MAOBI     0     1
4          Placebo     0     0
```

For placebo, the treatment label overlaps with the edges due to the x adjustment. We can easily change this value from 0 to 0.9.

```
R> nodes1 = nodes1 %>%
+    mutate(adj.x = replace(adj.x, trts == "Placebo", 0.9))
R> nodes1 %>% select(trts, adj.x, adj.y)
```

```
              trts adj.x adj.y
1            COMTI   0.0     0
2 Dopamine Agonist   1.0     1
3            MAOBI   0.0     1
4          Placebo   0.9     0
```

Now, we can use the modified adjustment settings in `nodes1` as input to argument `adj`; all other arguments are used to create a network graph similar to Figure 2.

```
R> netgraph(net1, plastic = FALSE, multiarm = TRUE,
+    iterate = TRUE, col = "black",
+    cex = 1.25, offset = ifelse(trts == "Placebo", 0.04, 0.02),
+    adj = nodes1 %>% select(adj.x, adj.y),
+    number.of.studies = TRUE, cex.number = 1)
```

The resulting network graph is shown in Figure 3.

Network meta-analysis results

The main NMA results are reported by printing `net1`.

```
R> net1

Number of studies: k = 29
Number of pairwise comparisons: m = 31
Number of observations: o = 5331
Number of treatments: n = 4
Number of designs: d = 4

Random effects model

Treatment estimate (sm = 'MD', comparison: other treatments vs 'Placebo'):
                    MD         95%-CI     z  p-value
COMTI            -0.95 [-1.27; -0.63] -5.79 < 0.0001
Dopamine Agonist -1.49 [-1.86; -1.12] -7.88 < 0.0001
MAOBI            -1.03 [-1.56; -0.50] -3.81   0.0001
Placebo              .              .     .        .

Quantifying heterogeneity / inconsistency:
tau^2 = 0.1518; tau = 0.3896; I^2 = 43.9% [12.3%; 64.2%]

Tests of heterogeneity (within designs) and inconsistency
(between designs):
                    Q d.f. p-value
Total           48.15   27  0.0074
Within designs  46.91   25  0.0050
Between designs  1.24    2  0.5376
```

The printout starts with a concise summary of the network: the number of studies, k, the number of pairwise comparisons, m, the number of observations, o, the number of treatments, n, and the number of designs, d.

The network estimates for the random effects models are provided next, taking placebo as reference treatment (as we used `reference.group = "plac"` to generate `net1`). We get network estimates for all treatments compared to placebo, even for treatments without any direct comparisons with placebo (which is not the case in `Stowe2010` as comparisons with placebo are available for all treatments, see Figure 3). Using argument `all.treatments =`

TRUE, we would get three matrices providing the random effects network estimates, lower and upper confidence limits for all observed and unobserved pairwise comparisons.

Recalling that the outcome is the difference in means of the off-time reduction of the first treatment minus the second treatment, the random effects NMA shows strong evidence that all adjuvant therapies, on a background of levodopa, reduced the off-time compared to placebo. Dopamine agonists seem to perform somewhat better than COMTI and MAOBI, with a average reduction of about 1.5 hours per day instead of 1 hour per day.

Information on heterogeneity and inconsistency is provided next. According to the Cochrane Handbook (Higgins *et al.* 2020), I^2 indicates a moderate level of heterogeneity / inconsistency. Furthermore, inconsistency between designs seems not to be an issue in this NMA.

The `summary.netmeta()` function generates a more detailed NMA printout including information on all pairwise comparisons from individual studies.

```
R> print(summary(net1), truncate = studlab %in% selstudy1, nchar.trts = 4,
+    nma = FALSE)

Original data (with adjusted standard errors for multi-arm studies):

                  treat1 treat2    TE  seTE seTE.adj narms multiarm
COMTI(E) INT-OZ     COMT   Plac -0.32 0.490    0.626     2
LARGO               COMT   Plac -0.80 0.212    0.543     3        *
LARGO               COMT   MAOB -0.02 0.212    0.543     3        *
LARGO               MAOB   Plac -0.78 0.212    0.543     3        *
COMTI(E) Nomecomt   COMT   Plac -1.20 0.356    0.528     2
*** Output truncated ***

Number of treatment arms (by study):
                  narms
COMTI(E) INT-OZ       2
LARGO                 3
COMTI(E) Nomecomt     2
*** Output truncated ***

Results (random effects model):

                  treat1 treat2    MD         95%-CI
COMTI(E) INT-OZ     COMT   Plac -0.95 [-1.27; -0.63]
LARGO               COMT   Plac -0.95 [-1.27; -0.63]
LARGO               COMT   MAOB  0.09 [-0.49;  0.67]
LARGO               MAOB   Plac -1.03 [-1.56; -0.50]
COMTI(E) Nomecomt   COMT   Plac -0.95 [-1.27; -0.63]
*** Output truncated ***


Legend:
 Abbreviation   Treatment name
```

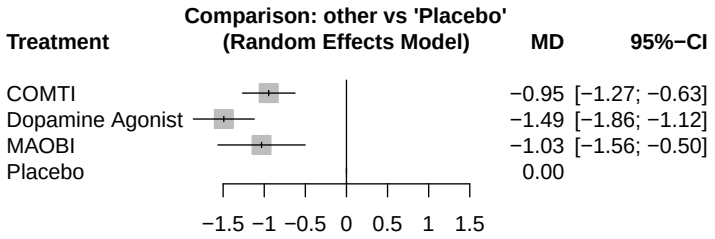


Figure 4: Forest plot for Parkinson's disease network meta-analysis (placebo as reference) (Stowe *et al.* 2010).

```
COMT            COMTI
Dopa Dopamine Agonist
MAOB            MAOBI
Plac          Placebo
```

We use the additional arguments to only show individual results for the previously selected three studies (argument `truncate`), abbreviate treatment labels to four characters (argument `nchar.trts`) and suppress the printing of the NMA results (argument `NMA`) which are identical to the printout of `net1`.

The first part of the detailed printout shows the original data in contrast-based format. The study labels and the first four columns (`treat1` to `seTE`) essentially come from `pw1`. We only notice that the third comparison of the three-arm study has been switched as treatments are sorted alpha-numerically by `netmeta()`. Accordingly, the mean difference switches from 0.78 (placebo versus MAOBI) to −0.78 (MAOBI versus placebo).

Next we see the adjusted standard errors (`seTE.adj`) which are identical for two-arm studies, however, inflated for the three-arm study. We can easily spot multi-arm studies in the printout as they are marked by an asterisk in column `multiarm`. Column `narms` reports the number of the treatment arms in each study which is also provided in a more concise way in the next part of the printout (labelled "Number of treatment arms (by study)").

The next part gives the network estimates under the random effects model for each pairwise comparison in the three studies. We notice that the network estimates and confidence intervals are the same for the three comparisons of COMTI and placebo. All network estimates and confidence intervals are identical to the printout of `net1`.

The legend at the bottom is printed as treatment labels are abbreviated. We could suppress printing of the legend using argument `legend = FALSE`.

Forest plots

A standard forest plot with placebo as reference shown in Figure 4 can be generated with one of the following commands.

```
R> forest(net1)
R> plot(net1)
```

A fuller picture is provided by the following command creating a forest plot showing all pairwise comparisons of all active treatments with other treatments (Figure 5). Of note,

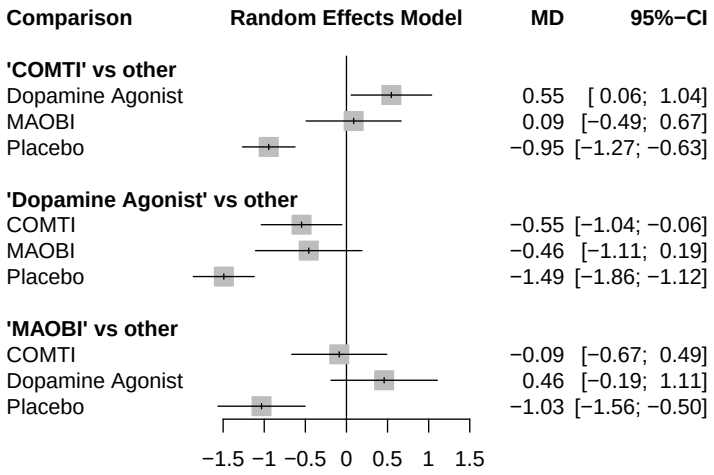


Figure 5: Forest plot for Parkinson's disease network meta-analysis (active treatment versus all other treatments) (Stowe *et al.* 2010).

the abbreviated treatment names provided to argument `reference.group` must be (and are) unambiguous.

```
R> forest(net1, ref = c("C", "D", "M"), baseline = FALSE, drop = TRUE)
```

Ranking of treatments

A treatment ranking using P-scores clearly shows that dopamine agonists appear to be the most effective treatment. The average proportion of treatments worse than dopamine agonists is about 97% under the random effects model.

```
R> netrank(net1)
```

```
                P-score
Dopamine Agonist  0.9677
MAOBI             0.5664
COMTI             0.4659
Placebo           0.0000
```

A more detailed picture of the treatment hierarchy is given by the rankograms for the four competing treatments.

```
R> set.seed(1909)
R> (ran1 <- rankogram(net1))
```

```
Rankogram (based on 1000 simulations)

Random effects model:
```

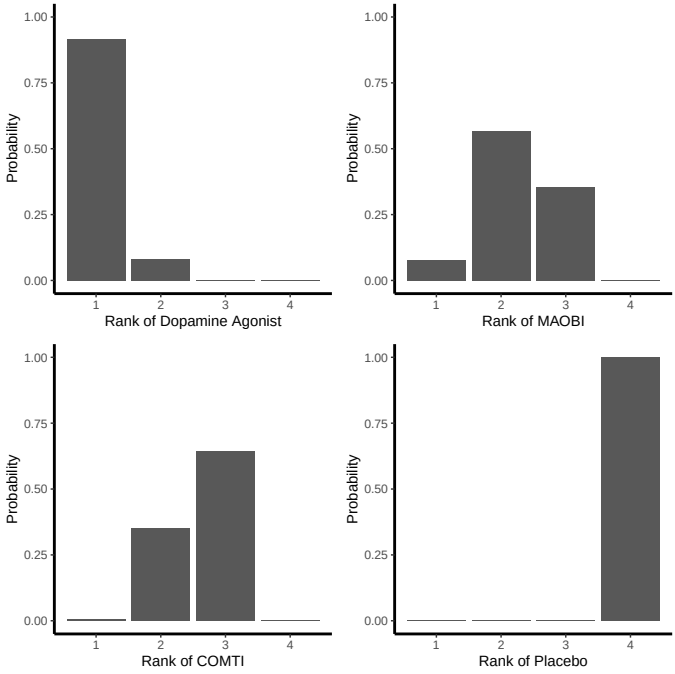


Figure 6: Rankogram for Parkinson's disease network meta-analysis (Stowe *et al.* 2010).

```
                    1      2      3      4
COMTI              0.0060 0.3520 0.6420 0.0000
Dopamine Agonist   0.9170 0.0800 0.0030 0.0000
MAOBI              0.0770 0.5680 0.3550 0.0000
Placebo            0.0000 0.0000 0.0000 1.0000
```

We use `set.seed()` to make resampling results reproducible. Dopamine agonists have 91.7% estimated probability of producing the best value compared to only 7.7% and 0.6% for the other active treatments. Placebo ranks last with 100% estimated probability. By default, 1000 simulations are conducted which could be changed in `rankogram()` using argument `nsim`.

The following command creates the rankogram shown in Figure 6.

```
R> plot(ran1)
```

As expected, SUCRA values based on 1000 simulations are very similar to P-scores reported before.

```
R> netrank(ran1)
```

```
                  SUCRA
Dopamine Agonist 0.9713
MAOBI            0.5740
COMTI            0.4547
Placebo          0.0000
```

```
- based on 1000 simulations
```

League table

A league table with random effects estimates (and without confidence limits to have a more concise printout in this article) sorted by decreasing P-scores is given by

```
R> netleague(net1, seq = netrank(net1), ci = FALSE)

League table (random effects model):

 Dopamine Agonist     .     .   -1.49
            -0.46 MAOBI  0.02   -1.03
            -0.55 -0.09 COMTI   -0.93
            -1.49 -1.03 -0.95 Placebo
```

Evaluation of heterogeneity and inconsistency

Next, we decompose Cochran's Q statistic to assess heterogeneity and inconsistency.

```
R> decomp.design(net1)

Q statistics to assess homogeneity / consistency

                    Q df p-value
Total           48.15 27  0.0074
Within designs  46.91 25  0.0050
Between designs  1.24  2  0.5376

Design-specific decomposition of within-designs Q statistic

                   Design     Q df p-value
            Placebo:COMTI 23.65 10  0.0086
 Placebo:Dopamine Agonist 21.70 14  0.0850
            Placebo:MAOBI  1.56  1  0.2114

Between-designs Q statistic after detaching of single designs

     Detached design    Q df p-value
       Placebo:COMTI 1.21  1  0.2709
       Placebo:MAOBI 0.02  1  0.8793
 Placebo:COMTI:MAOBI 0.00  0      --

Q statistic to assess consistency under the assumption of
a full design-by-treatment interaction random effects model
```

```
                  Q df p-value tau.within tau2.within
Between designs 0.45  2  0.7968     0.4547      0.2067
```

The first part was already reported in the printout of `net1`. The design-specific decomposition of Q_W shows that the within-design heterogeneity can largely be traced back to the comparison of COMTI versus placebo. Results for detaching single designs and the full design-by-treatment interaction model do not show evidence of inconsistency between designs.

Inconsistency between direct and indirect evidence within each network estimate can be evaluated using the SIDE method.

```
R> netsplit(net1)
```

```
Separate indirect from direct evidence (SIDE) using back-calculation method

Random effects model:

               comparison  k prop   nma direct indir.  Diff     z p-value
   COMTI:Dopamine Agonist  0    0  0.55      .   0.55     .     .       .
              COMTI:MAOBI  1 0.44  0.09  -0.02   0.17 -0.19 -0.33  0.7450
            COMTI:Placebo 12 0.98 -0.95  -0.93  -1.65  0.72  0.66  0.5071
   Dopamine Agonist:MAOBI  0    0 -0.46      .  -0.46     .     .       .
 Dopamine Agonist:Placebo 15 1.00 -1.49  -1.49      .     .     .       .
            MAOBI:Placebo  3 0.90 -1.03  -1.03  -1.08  0.05  0.06  0.9544

Legend:
 comparison - Treatment comparison
 k          - Number of studies providing direct evidence
 prop       - Direct evidence proportion
 nma        - Estimated treatment effect (MD) in network meta-analysis
 direct     - Estimated treatment effect (MD) derived from direct evidence
 indir.     - Estimated treatment effect (MD) derived from indirect evidence
 Diff       - Difference between direct and indirect treatment estimates
 z          - z-value of test for disagreement (direct versus indirect)
 p-value    - p-value of test for disagreement (direct versus indirect)
```

The printout shows that there are three pairwise comparisons contributing both direct and indirect evidence, for none of which there is evidence of inconsistency. Furthermore, COMTI versus placebo and MAOBI versus placebo are mainly informed by direct evidence (column `prop`).

Finally, the net heat plot also does not show any evidence of disagreement between direct and indirect estimates (Figure 7).

```
R> netheat(net1, nchar.trts = 4)
```

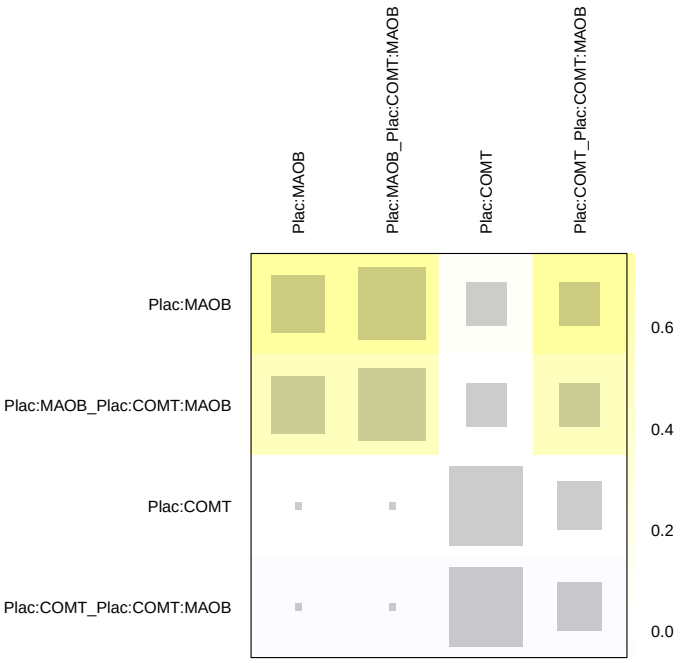


Figure 7: Net heat plot for Parkinson's disease network meta-analysis (Stowe *et al.* 2010).

5.2. Example 2: Antithrombotics to prevent strokes

We change the appearance of confidence intervals for the next two examples, which both use the odds ratio as effect measure.

```
R> cilayout("(", "-")
```

First we transform `Dogliotti2014` to contrast-based format.

```
R> pw2 <- pairwise(treat = treatment, n = total, event = stroke,
+    studlab = study, data = Dogliotti2014, sm = "OR")
```

This command is more concise compared to the first example using `list()` as the pivotal arguments relate to single variables in a data set in long arm-based format. For binary outcomes, we have to specify sample sizes (argument `n`) and the number of events (`event`) in addition to treatment and study labels. The odds ratio is used as effect measure (while the risk ratio is the default).

In R object `pw2`, variable `TE` contains the log odds ratio and `seTE` its standard error.

```
R> head(pw2, 5) %>% select(studlab, treat1, treat2, TE, seTE)
```

```
        studlab  treat1          treat2         TE      seTE
1 AFASAK-I 1989    VKAs         Aspirin -0.5939405 0.4240325
2 AFASAK-I 1989    VKAs Placebo/Control -0.7752100 0.4122678
3 AFASAK-I 1989 Aspirin Placebo/Control -0.1812695 0.3484410
4   BAATAF 1990    VKAs Placebo/Control -1.5356718 0.6482047
5     CAFA 1991    VKAs Placebo/Control -0.3999555 0.5373985
```

Common and random effects NMAs using the inverse variance method can be conducted in the usual way.

```
R> net2 <- netmeta(pw2, ref = "plac")

Comparison not considered in network meta-analysis:
     studlab treat1  treat2 TE seTE
 WASPO, 2007   VKAs Aspirin NA   NA
```

We get a warning that the WASPO study is excluded from the NMA. We can confirm that the study is excluded due to zero events in both treatment groups.

```
R> subset(pw2, studlab == "WASPO, 2007") %>%
+    select(studlab, event1, n1, event2, n2)

       studlab event1 n1 event2 n2
28 WASPO, 2007      0 36      0 39
```

The network graph shown in Figure 8 was generated with the following command.

```
R> netgraph(net2, seq = "optimal",
+    plastic = FALSE, number.of.studies = TRUE, cex.points = n.trts,
+    offset =
+      ifelse(trts == "VKAs", 0.05,
+      ifelse(trts %in% c("Aspirin", "Apixaban"), 0.035, 0.025)),
+    labels = paste0(trts, "\n(n=", round(n.trts), ")"))
```

We notice that the majority of studies compared Aspirin, vitamin K antagonists (VKAs) and Placebo/Control. Furthermore, the most commonly administered treatments are VKAs, Apixaban and Aspirin.

As the example concerns a rare binary outcome, we also perform NMA using the Mantel-Haenszel approach and we compare the results with the inverse variance (conventional) NMA.

```
R> net2.mh <- netmetabin(pw2, ref = "plac")
```

We use `netbind()` to create a forest plot with the common effects estimates from the two models.

```
R> nb2 <- netbind(net2, net2.mh, random = FALSE,
+    name = c("Inverse variance", "Mantel-Haenszel"))
R> forest(nb2, xlim = c(0.15, 2), at = c(0.2, 0.5, 1, 2))
```

The two analyses lead to very similar results (Figure 9). Accordingly, we would proceed with the inverse variance method which has the advantage that it also provides results for a random effects NMA.

Irrespective of whether we use the inverse variance (`net2`) or Mantel-Haenszel (`net2.mh`) method, the same workflow as for example 1 can be followed:

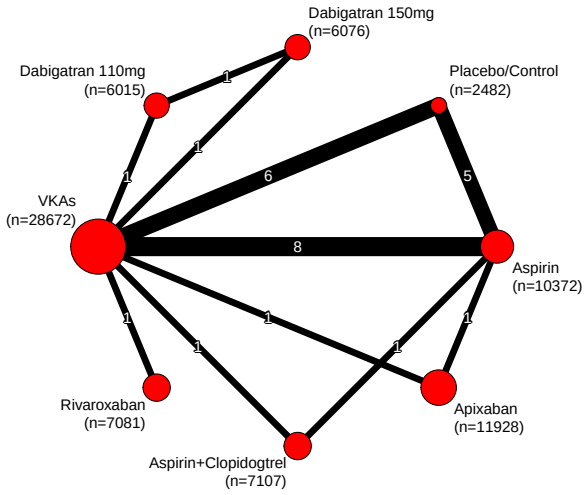


Figure 8: Network graph for antithrombotics network meta-analysis (Dogliotti *et al.* 2014). Line widths are proportional to the number of studies directly comparing treatments. Point sizes are proportional to the number of observations in treatment arms.

- `netgraph()` to generate a network graph,
- `print.netmeta()` or `summary.netmeta()` to print the results,
- `forest()` to create a forest plot,
- `netleague()` to generate a league table,
- `netrank()` and `rankogram()` to rank treatments,
- `decomp.design()`, `netsplit()`, and `netheat()` to evaluate heterogeneity and inconsistency.

5.3. Example 3: Treatments for chronic obstructive pulmonary disease

We start the analysis of the third example data set by pre-processing the data for NMA.

```
R> pw3 <- pairwise(treat = treatment, n = total, event = exac,
+    studlab = paste(study, year), data = Baker2009, sm = "OR")
R> net3 <- netmeta(pw3, common = FALSE, ref = "plac")

Comparisons not considered in network meta-analysis:
            studlab                 treat1     treat2 TE seTE
 DalNegro 2003 2008 Fluticasone+Salmeterol    Placebo NA   NA
 DalNegro 2003 2008 Fluticasone+Salmeterol Salmeterol NA   NA
 DalNegro 2003 2008             Salmeterol    Placebo NA   NA
```

We notice that the excluded DalNegro study is very small with events for all participants.

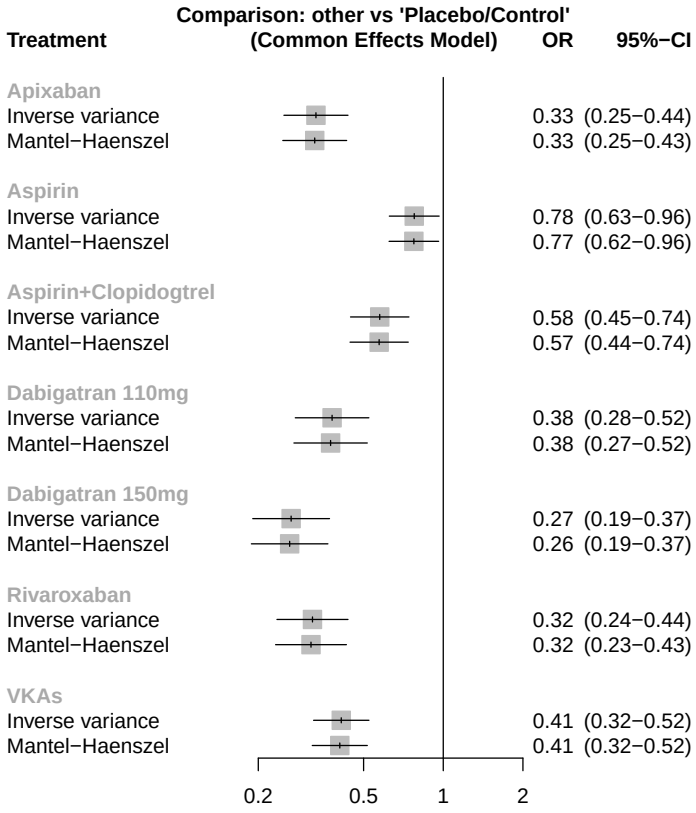


Figure 9: Forest plot comparing results for inverse variance and Mantel-Haenszel method in antithrombotics network meta-analysis (Dogliotti *et al.* 2014).

```
R> subset(pw3, studlab == "DalNegro 2003 2008") %>%
+    select(studlab, event1, n1, event2, n2)

              studlab event1 n1 event2 n2
75 DalNegro 2003 2008      6  6      6  6
76 DalNegro 2003 2008      6  6      6  6
77 DalNegro 2003 2008      6  6      6  6
```

A comparison of inverse variance and Mantel-Haenszel NMA reveals that common effects estimates are again very similar (results not shown).

The network graph in Figure 10 was generated with the following command. We wanted the placebo node to be on the right so we use argument `rotate` to turn the network plot three positions counter-clockwise ($-3 \cdot 360°/8$ treatments $= -3 \cdot 45°$).

```
R> netgraph(net3, seq = "optimal",
+    plastic = FALSE, number.of.studies = TRUE, cex.points = n.trts,
+    offset = ifelse(n.trts < 1500, 0.025, 0.05),
+    labels = paste0(gsub("+", " +\n", trts, fixed = TRUE),
```

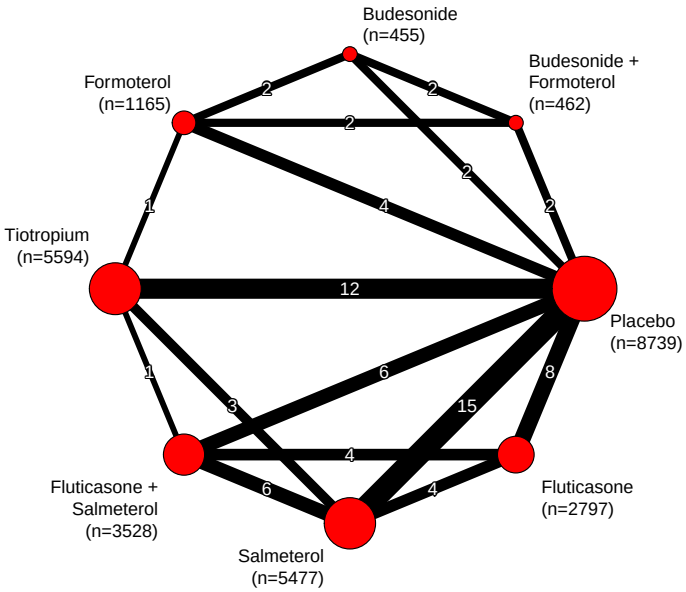


Figure 10: Network graph for COPD network meta-analysis (Baker *et al.* 2009). Line widths are proportional to the number of studies directly comparing treatments. Point sizes are proportional to the number of observations in treatment arms.

```
+      "\n(n=", round(n.trts), ")"),
+    rotate = -3 * 45)
```

We notice that the network is well connected and that several studies evaluated the combination fluticasone + salmeterol.

The standard NMA shows that all treatments but budesonide and formoterol are more effective than placebo. Furthermore, the combination treatments are more effective than the individual treatments.

```
R> net3

Number of studies: k = 38
Number of pairwise comparisons: m = 74
Number of observations: o = 28217
Number of treatments: n = 8
Number of designs: d = 13

Random effects model

Treatment estimate (sm = 'OR', comparison: other treatments vs 'Placebo'):
                         OR      95%-CI     z  p-value
Budesonide             0.80 (0.59-1.10) -1.37   0.1702
Budesonide+Formoterol  0.67 (0.49-0.90) -2.63   0.0086
Fluticasone            0.86 (0.75-0.99) -2.05   0.0406
Fluticasone+Salmeterol 0.75 (0.66-0.86) -4.17 < 0.0001
```

```
Formoterol             1.00 (0.80-1.25) -0.00   1.0000
Placebo                   .           .     .        .
Salmeterol             0.80 (0.72-0.89) -4.06 < 0.0001
Tiotropium             0.69 (0.61-0.77) -6.39 < 0.0001

Quantifying heterogeneity / inconsistency:
tau^2 = 0.0111; tau = 0.1055; I^2 = 24.4% (0.0%-47.7%)

Tests of heterogeneity (within designs) and inconsistency
(between designs):
                    Q d.f. p-value
Total           60.86   46  0.0700
Within designs  43.20   34  0.1340
Between designs 17.66   12  0.1264
```

The next step is based on the assumption that the addition of placebo in any of the other treatments in the network does not change the risk of COPD exacerbations (we specify this through the argument `inactive`). (We note in parentheses that it is not necessary in general to specify an inactive treatment. For example, this is not possible if there are only active interventions, or if an active effect must be assumed also for a reference intervention such as "treatment as usual"). Assuming placebo as inactive, we now conduct a CNMA. We do not have to specify the separator (argument `sep.comps`), as the plus sign used in treatment labels of `Baker2009` is the default.

```
R> (nc3 <- netcomb(net3, inactive = "plac"))

Number of studies: k = 38
Number of pairwise comparisons: m = 74
Number of treatments: n = 8
Number of active components: c = 5
Number of designs: d = 13

Results for combinations (additive model, random effects model):
                         OR      95%-CI     z  p-value
Budesonide             0.73 (0.58-0.92) -2.64   0.0084
Budesonide+Formoterol  0.69 (0.52-0.92) -2.51   0.0121
Fluticasone            0.90 (0.81-1.00) -2.06   0.0397
Fluticasone+Salmeterol 0.74 (0.65-0.84) -4.58 < 0.0001
Formoterol             0.94 (0.78-1.14) -0.62   0.5356
Placebo                1.00 (1.00-1.00)    --       --
Salmeterol             0.82 (0.75-0.90) -4.28 < 0.0001
Tiotropium             0.69 (0.61-0.77) -6.45 < 0.0001

Results for components (random effects model):
              OR      95%-CI     z  p-value
Budesonide  0.73 (0.58-0.92) -2.64   0.0084
Fluticasone 0.90 (0.81-1.00) -2.06   0.0397
```

```
Formoterol  0.94 (0.78-1.14) -0.62   0.5356
Salmeterol  0.82 (0.75-0.90) -4.28 < 0.0001
Tiotropium  0.69 (0.61-0.77) -6.45 < 0.0001

Quantifying heterogeneity / inconsistency:
tau^2 = 0.0101; tau = 0.1003; I^2 = 23.5% (0.0%-46.7%)

Heterogeneity statistics:
                   Q df p-value
Additive model 62.74 48  0.0750
Standard model 60.86 46  0.0700
Difference      1.88  2  0.3908
```

The number of active components is printed in addition to the other well known quantities, followed by results for treatment combinations and components. The estimated odds ratio for the combination fluticasone + salmeterol is equal to the product of the component odds ratios: $0.74 = 0.90 \cdot 0.82$. Equivalently, the log odds ratio of the combination is equal to the sum of the component log odds ratios. A comparison of the heterogeneity statistics shows that the additive model fits the data nearly as well as the standard NMA. The Q test under "difference" corresponds to a likelihood-ratio test (Rücker *et al.* 2021).

We use `netbind()` again to produce a forest plot with results from standard NMA and additive CNMA. Overall, the random effects estimates of standard NMA and additive CNMA do not differ much (Figure 11).

```
R> nb3 <- netbind(net3, nc3, name = c("Standard NMA", "Additive CNMA"))
R> forest(nb3)
```

In a further step, we can rank treatments based on the treatment estimates from the additive CNMA.

```
R> netrank(nc3)
```

```
                       P-score
Tiotropium              0.8588
Budesonide+Formoterol   0.8161
Fluticasone+Salmeterol  0.7038
Budesonide              0.6909
Salmeterol              0.4460
Fluticasone             0.2649
Formoterol              0.1770
Placebo                 0.0426
```

This CNMA ranking shows some differences to the ranking based on the standard NMA.

```
R> netrank(net3)
```

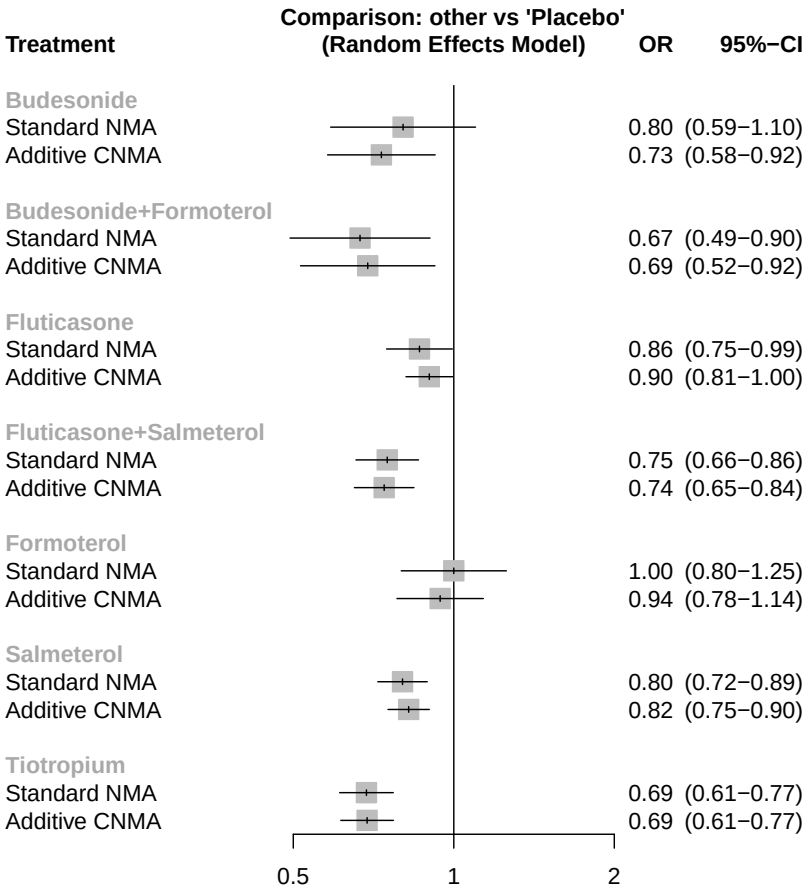


Figure 11: Forest plot comparing results for NMA and additive CNMA in COPD network meta-analysis (Baker *et al.* 2009).

```
                      P-score
Tiotropium             0.8719
Budesonide+Formoterol  0.8539
Fluticasone+Salmeterol 0.6856
Budesonide             0.5199
Salmeterol             0.5162
Fluticasone            0.3523
Formoterol             0.1132
Placebo                0.0871
```

Overall, the ranking order does not change when considering an additive CNMA instead of a standard NMA.

R functions `decomp.design()`, `netsplit()`, and `netheat()` to assess heterogeneity and inconsistency are not available for CNMAs.

6. Conclusions and outlook

This article provides a general introduction and overview to R package **netmeta** for frequentist NMA. A central aim of **netmeta** is to provide methods for all essential steps in an NMA in a user-friendly way.

The main hurdle in **netmeta** is to transform the available data set into the comparison-based format required by `netmeta()` for standard NMA, `netmetabin()` for NMA with rare binary outcomes, and `discomb()` for disconnected networks. Accordingly, use of R function `pairwise()` constitutes the main pre-processing step. Our examples demonstrate its use for the most common NMA data formats, i.e., wide arm-based and long arm-based format. After that, the typical (C)NMA workflow as outlined in this article is straightforward to follow, after conducting the main analysis which is the input to all subsequent analyses.

The simplicity of **netmeta** has led to its use by two web-applications, MetaInsight and CINeMA (confidence in network meta-analysis) (Owen *et al.* 2019; Nikolakopoulou *et al.* 2020).

While **netmeta** offers a comprehensive set of functions for NMA and an easy to use interface, it lacks on flexibility compared to the more general R packages **metafor** (Viechtbauer 2010) and **mixmeta** (Sera *et al.* 2019). A limitation of **netmeta** is that methods to conduct subgroup analysis and network meta-regression are not readily available. Moreover, further levels of hierarchical modelling cannot be added, in contrast with Bayesian models. Accordingly, possible future developments include providing subgroup analysis and meta-regression and linking one or more R packages for Bayesian NMA with **netmeta** in order to make more flexible NMA methods available to applied scientists in a user-friendly way.

In summary, **netmeta** offers a wide range of analysis and presentational tools for fitting frequentist NMA in R. This article presents the main capabilities of **netmeta**; however, the package is constantly updated with new features and methods in line with methodological research and new needs that arise when conducting an NMA.

Acknowledgments

OE is supported by the Swiss National Science Foundation (SNSF), grant 180083. GS and TP are supported by project funding (Grant No. 179158) from the SNSF. AN is supported by a SNSF personal fellowship (P400PM_186723). GR was supported by the German Research Foundation (DFG), grant RU 1747/1-2.

*REFERENCES

Albert AE (1972). *Regression and the Moore-Penrose Pseudoinverse*. Mathematics in Science and Engineering. Academic Press, New York. doi:10.1016/s0076-5392(08)x6167-3.

Bailey RA (2007). "Designs for Two-Colour Microarray Experiments." *Journal of the Royal Statistical Society C*, **56**(4), 365–394. doi:10.1111/j.1467-9876.2007.00582.x.

Baker WL, Baker EL, Coleman CI (2009). "Pharmacologic Treatments for Chronic Obstructive Pulmonary Disease: A Mixed-Treatment Comparison Meta-Analysis." *Pharmacotherapy: The Journal of Human Pharmacology and Drug Therapy*, **29**(8), 891–905. doi:10.1592/phco.29.8.891.

Balduzzi S, Rücker G, Schwarzer G (2019). "How to Perform a Meta-Analysis with R: A Practical Tutorial." *Evidence-Based Mental Health*, **22**(4), 153–160. doi:10.1136/ebmental-2019-300117.

Béliveau A, Boyne DJ, Slater J, Brenner D, Arora P (2019). "**BUGSnet**: An R Package to Facilitate the Conduct and Reporting of Bayesian Network Meta-Analyses." *BMC Medical Research Methodology*, **19**(1), 196. doi:10.1186/s12874-019-0829-2.

Carlsen L, Bruggemann R (2014). "Partial Order Methodology: A Valuable Tool in Chemometrics." *Journal of Chemometrics*, **28**(4), 226–234. doi:10.1002/cem.2569.

Chaimani A, Higgins JP, Mavridis D, Spyridonos P, Salanti G (2013). "Graphical Tools for Network Meta-Analysis in Stata." *PLOS One*, **8(10)**(10), e76654. doi:10.1371/journal.pone.0076654.

Chaimani A, Salanti G (2012). "Using Network Meta-Analysis to Evaluate the Existence of Small-Study Effects in a Network of Interventions." *Research Synthesis Methods*, **3**(2), 161–176. doi:10.1002/jrsm.57.

Crippa A, Orsini N (2016). "Dose-Response Meta-Analysis of Differences in Means." *BMC Medical Research Methodology*, **16**, 91. doi:10.1186/s12874-016-0189-0.

Csárdi G (2019). ***cranlogs**: Download Logs from the RStudio CRAN Mirror*. R package version 2.1.1, URL https://CRAN.R-project.org/package=cranlogs.

Davies AL, Papakonstantinou T, Nikolakopoulou A, Rücker G, Galla T (2022). "Network Meta-Analysis and Random Walks." *Statistics in Medicine*, **41**(12), 2091–2114. doi:10.1002/sim.9346.

Dias S, Welton NJ, Caldwell DM, Ades AE (2010). "Checking Consistency in Mixed Treatment Comparison Meta-Analysis." *Statistics in Medicine*, **29**(7-8), 932–944. doi:10.1002/sim.3767.

Dogliotti A, Paolasso E, Giugliano RP (2014). "Current and New Oral Antithrombotics in Nonvalvular Atrial Fibrillation: A Network Meta-Analysis of 79 808 Patients." *Heart*, **100**, 396–405. doi:10.1136/heartjnl-2013-304347.

Efthimiou O, Debray TPA, van Valkenhoef G, Trelle S, Panayidou K, Moons KGM, Reitsma JB, Shangg A, Salanti G, GetReal Methods Review Group (2015). "GetReal in Network Meta-Analysis: A Review of the Methodology." *Research Synthesis Methods*, **7**(3), 236–263. doi:10.1002/jrsm.1195.

Efthimiou O, Rücker G, Schwarzer G, Higgins JPT, Egger M, Salanti G (2019). "Network Meta-Analysis of Rare Events Using the Mantel-Haenszel Method." *Statistics in Medicine*, **38**(16), 2992–3012. doi:10.1002/sim.8158.

Greenland S (1989). "Generalized Mantel-Haenszel Estimators for $K2\times J$ Tables." *Biometrics*, **45**(1), 183–191. doi:10.2307/2532044.

Guenhan BK, Friede T, Held L (2018). "A Design-by-Treatment Interaction Model for Network Meta-Analysis and Meta-Regression with Integrated Nested Laplace Approximations." *Research Synthesis Methods*, **9**(2), 179–194. doi:10.1002/jrsm.1285.

Gutman I, Xiao W (2004). "Generalized Inverse of the Laplacian Matrix and Some Applications." *Bulletin De l'Académie Serbe Des Sciences Et Des Arts*, **129**(29), 15–23. doi:10.2298/bmat0429015g.

Higgins JPT, Jackson D, Barrett JK, Lu G, Ades AE, White IR (2012). "Consistency and Inconsistency in Network Meta-Analysis: Concepts and Models for Multi-Arm Studies." *Research Synthesis Methods*, **3**(2), 98–110. doi:10.1002/jrsm.1044.

Higgins JPT, Thomas J, Chandler J, Cumpston M, Li T, Page MJ, Welch VA (2020). *Cochrane Handbook for Systematic Reviews of Interventions Version 6.1*. Cochrane. doi:10.1002/9781119536604.

Higgins JPT, Thompson SG (2002). "Quantifying Heterogeneity in a Meta-Analysis." *Statistics in Medicine*, **21**(11), 1539–1558. doi:10.1002/sim.1186.

Hu Y (2012). "Algorithms for Visualizing Large Networks." In U Naumann, O Schenk (eds.), *Combinatorial Scientific Computing*, pp. 525–549. Chapman & Hall, Boca Raton.

Hutton B, Salanti G, Caldwell DM, Chaimani A, Schmid CH, Cameron C, Ioannidis JPA, Straus S, Thorlund K, Jansen JP, Mulrow C, Catalá-López F, Goetzsche PC, Dickersin K, Boutron I, Altman DG, Moher D (2015). "The PRISMA Extension Statement for Reporting of Systematic Reviews Incorporating Network Meta-Analyses of Health Care Interventions: Checklist and Explanations." *Annals of Internal Medicine*, **162**, 777–784. doi:10.7326/m14-2385.

Jackson D, Riley R, White IR (2011). "Multivariate Meta-Analysis: Potential and Promise." *Statistics in Medicine*, **30**(20), 2481–2498. doi:10.1002/sim.4172.

Jackson D, White IR, Riley RD (2012). "Quantifying the Impact of Between-Study Heterogeneity in Multivariate Meta-Analyses." *Statistics in Medicine*, **31**(29), 3805–3820. doi:10.1002/sim.5453.

Jackson D, White IR, Riley RD (2013). "A Matrix-Based Method of Moments for Fitting the Multivariate Random Effects Model for Meta-Analysis and Meta-Regression." *Biometrical Journal*, **55**(2), 231–245. doi:10.1002/bimj.201200152.

Kamada T, Kawai S (1989). "An Algorithm for Drawing General Undirected Graphs." *Information Processing Letters*, **31**(1), 7–15. doi:10.1016/0020-0190(89)90102-6.

Krahn U, Binder H, König J (2013). "A Graphical Tool for Locating Inconsistency in Network Meta-Analyses." *BMC Medical Research Methodology*, **13**, 35. doi:10.1186/1471-2288-13-35.

König J, Krahn U, Binder H (2013). "Visualizing the Flow of Evidence in Network Meta-Analysis and Characterizing Mixed Treatment Comparisons." *Statistics in Medicine*, **32**(30), 5414–5429. doi:10.1002/sim.6001.

Lin L, Zhang J, Hodges JS, Chu H (2020). "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.

Mantel N, Haenszel W (1959). "Statistical Aspects of the Analysis of Data from Retrospective Studies of Disease." *Journal of the National Cancer Institute*, **22**(4), 719–748. doi:10.1093/jnci/22.4.719.

Nikolakopoulou A, Higgins JPT, Papakonstantinou T, Chaimani A, Del Giovane C, Egger M, Salanti G (2020). "CINeMA: An Approach for Assessing Confidence in the Results of a Network Meta-Analysis." *PLOS Medicine*, **17**(4), 1–19. doi:10.1371/journal.pmed.1003082.

Owen RK, Bradbury N, Xin Y, Cooper N, Sutton A (2019). "MetaInsight: An Interactive Web-Based Tool for Analyzing, Interrogating, and Visualizing Network Meta-Analyses Using R-Shiny and **netmeta**." *Research Synthesis Methods*, **10**(4), 569–581. doi:10.1002/jrsm.1373.

Papakonstantinou T (2019). ***nmadb**: Network Meta-Analysis Database API*. R package version 1.2.0, URL https://CRAN.R-project.org/package=nmadb.

Papakonstantinou T, Nikolakopoulou A, Rücker G, Chaimani A, Schwarzer G, Egger M, Salanti G (2018). "Estimating the Contribution of Studies in Network Meta-Analysis: Paths, Flows and Streams." *F1000Research*, **7**(610). doi:10.12688/f1000research.14770.3.

Paterson L (1983). "Circuits and Efficiency in Incomplete Block Designs." *Biometrika*, **70**(1), 215–225. doi:10.1093/biomet/70.1.215.

Petropoulou M, Nikolakopoulou A, Veroniki AA, Rios P, Vafaei A, Zarin W, Giannatsi M, Sullivan S, Tricco AC, Chaimani A, Egger M, Salanti G (2017). "Bibliographic Study Showed Improving Statistical Methodology of Network Meta-Analyses Published Between 1999 and 2015." *Journal of Clinical Epidemiology*, **82**, 20–28. doi:10.1016/j.jclinepi.2016.11.002.

Phillippo DM (2022). ***multinma**: Network Meta-Analysis of Individual and Aggregate Data in Stan*. R package version 0.5.0, URL https://CRAN.R-project.org/package=multinma.

Rao CR, Mitra SK (1971). *Generalized Inverse of Matrices and Its Applications*. John Wiley & Sons, New York, London, Sydney, Toronto.

Röver C (2020). "Bayesian Random-Effects Meta-Analysis Using the **bayesmeta** R Package." *Journal of Statistical Software*, **93**(6), 1–51. doi:10.18637/jss.v093.i06.

Rücker G (2012). "Network Meta-Analysis, Electrical Networks and Graph Theory." *Research Synthesis Methods*, **3**(4), 312–324. doi:10.1002/jrsm.1058.

Rücker G, Krahn U, König J, Efthimiou O, Davies A, Papakonstantinou T, Schwarzer G (2022). ***netmeta**: Network Meta-Analysis Using Frequentist Methods*. R package version 2.7-0, URL https://CRAN.R-project.org/package=netmeta.

Rücker G, Nikolakopoulou A, Papakonstantinou T, Salanti G, Riley RD, Schwarzer G (2020a). "The Statistical Importance of a Study for a Network Meta-Analysis Estimate." *BMC Medical Research Methodology*, **20**(1), 190. doi:10.1186/s12874-020-01075-y.

Rücker G, Petropoulou M, Schwarzer G (2020b). "Network Meta-Analysis of Multicomponent Interventions." *Biometrical Journal*, **62**(3), 808–821. doi:10.1002/bimj.201800167.

Rücker G, Schmitz S, Schwarzer G (2021). "Component Network Meta-Analysis Compared to a Matching Method in a Disconnected Network: A Case Study." *Biometrical Journal*, **63**(2), 447–461. doi:10.1002/bimj.201900339.

Rücker G, Schwarzer G (2014). "Reduce Dimension or Reduce Weights? Comparing Two Approaches to Multi-Arm Studies in Network Meta-Analysis." *Statistics in Medicine*, **33**, 4353–4369. doi:10.1002/sim.6236.

Rücker G, Schwarzer G (2015). "Ranking Treatments in Frequentist Network Meta-Analysis Works Without Resampling Methods." *BMC Medical Research Methodology*, **15**, 58. doi:10.1186/s12874-015-0060-8.

Rücker G, Schwarzer G (2016). "Automated Drawing of Network Plots in Meta-Analysis." *Research Synthesis Methods*, **7**(1), 94–107. doi:10.1002/jrsm.1143.

Rücker G, Schwarzer G (2017). "Resolve Conflicting Rankings of Outcomes in Network Meta-Analysis: Partial Ordering of Treatments." *Research Synthesis Methods*, **8**(4), 526–536. doi:10.1002/jrsm.1270.

Salanti G (2012). "Indirect and Mixed-Treatment Comparison, Network, or Multiple-Treatments Meta-Analysis: Many Names, Many Benefits, Many Concerns for the Next Generation Evidence Synthesis Tool." *Research Synthesis Methods*, **3**(2), 80–97. doi:10.1002/jrsm.1037.

Salanti G, Ades AE, Ioannidis JPA (2011). "Graphical Methods and Numerical Summaries for Presenting Results from Multiple-Treatment Meta-Analysis: An Overview and Tutorial." *Journal of Clinical Epidemiology*, **64**(2), 163–171. doi:10.1016/j.jclinepi.2010.03.016.

Salanti G, Higgins JP, Ades AE, Ioannidis JP (2008). "Evaluation of Networks of Randomized Trials." *Statistical Methods in Medical Research*, **17**(3), 279–301. doi:10.1177/0962280207080643.

Schwarzer G (2007). "**meta**: An R Package for Meta-Analysis." *R News*, **7**(3), 40–45.

Schwarzer G, Carpenter JR, Rücker G (2015). *Meta-Analysis with R*. Use R! Springer-Verlag, Switzerland. doi:10.1007/978-3-319-21416-0.

Senn S, Gavini F, Magrez D, Scheen A (2013). "Issues in Performing a Network Meta-Analysis." *Statistical Methods in Medical Research*, **22**(2), 169–189. doi:10.1177/0962280211432220.

Seo M, Schmid C (2022). ***bnma**: Bayesian Network Meta-Analysis Using **JAGS***. R package version 1.5.0, URL https://CRAN.R-project.org/package=bnma.

Sera F, Armstrong B, Blangiardo M, Gasparrini A (2019). "An Extended Mixed-Effects Framework for Meta-Analysis." *Statistics in Medicine*, **38**, 5429–5444. doi:10.1002/sim.8362.

Stijnen T, Hamza TH, Ozdemir P (2010). "Random Effects Meta-Analysis of Event Outcome in the Framework of the Generalized Linear Mixed Model with Applications in Sparse Data." *Statistics in Medicine*, **29**(29), 3046–3067. doi:10.1002/sim.4040.

Stowe R, Ives N, Clarke CE, Deane K, van Hilten, Wheatley K, Gray R, Handley K, Furmston A (2010). "Evaluation of the Efficacy and Safety of Adjuvant Treatment to Levodopa Therapy in Parkinson's Disease Patients with Motor Complications." *Cochrane Database of Systematic Reviews*. doi:10.1002/14651858.cd007166.pub2.

van Valkenhoef G, Kuiper J (2021). ***gemtc**: Network Meta-Analysis Using Bayesian Methods*. R package version 1.0-1, URL https://CRAN.R-project.org/package=gemtc.

Viechtbauer W (2010). "Conducting Meta-Analyses in R with the **metafor** Package." *Journal of Statistical Software*, **36**(3), 1–48. doi:10.18637/jss.v036.i03.

Welton NJ, Caldwell DM, Adamopoulos E, Vedhara K (2009). "Mixed Treatment Comparison Meta-Analysis of Complex Interventions: Psychological Interventions in Coronary Heart Disease." *American Journal of Epidemiology*, **169**(9), 1158–1165. doi:10.1093/aje/kwp014.

White IR (2015). "Network Meta-Analysis." *Stata Journal*, **15**, 951–985. doi:10.1177/1536867x1501500403.

Wickham H, François R, Henry L, Müller K (2022). ***dplyr**: A Grammar of Data Manipulation*. R package version 1.0.10, URL https://CRAN.R-project.org/package=dplyr.

Yates F (1940). "The Recovery of Inter-Block Information in Balanced Incomplete Block Designs." *Annals of Eugenics*, **10**(4), 317–325. doi:10.1111/j.1469-1809.1940.tb02257.x.

Zarin W, Veroniki AA, Nincic V, Vafaei A, Reynen E, Motiwala SS, Antony J, Sullivan SM, Rios P, Daly C, Ewusie J, Petropoulou M, Nikolakopoulou A, Chaimani A, Salanti G, Straus SE, Tricco AC (2017). "Characteristics and Knowledge Synthesis Approach for 456 Network Meta-Analyses: A Scoping Review." *BMC Medicine*, **15**(1), 3. doi:10.1186/s12916-016-0764-6.

Affiliation:

Sara Balduzzi, Gerta Rücker, Adriani Nikolakopoulou, Theodoros Papakonstantinou, Guido Schwarzer
Institute of Medical Biometry and Statistics
Faculty of Medicine and Medical Center
University of Freiburg
Stefan-Meier-Strasse 26
79104 Freiburg, Germany
E-mail: balduzzi.sara@gmail.com
gerta.ruecker@uniklinik-freiburg.de
adriani.nikolakopoulou@uniklinik-freiburg.de
theodoros.papakonstantinou@uniklinik-freiburg.de
guido.schwarzer@uniklinik-freiburg.de
URL: https://www.uniklinik-freiburg.de/imbi.html

Georgia Salanti
Institute of Social and Preventive Medicine
University of Bern
Mittelstrasse 43
3012 Bern, Switzerland
E-mail: georgia.salanti@ispm.unibe.ch
URL: https://www.ispm.unibe.ch

Orestis Efthimiou
Institute of Social and Preventive Medicine
University of Bern
Mittelstrasse 43
3012 Bern, Switzerland
and
Department of Psychiatry
University of Oxford
Oxford, United Kingdom
E-mail: orestis.efthimiou@ispm.unibe.ch
URL: https://www.ispm.unibe.ch

Journal of Statistical Software https://www.jstatsoft.org/
published by the Foundation for Open Access Statistics https://www.foastat.org/
March 2023, Volume 106, Issue 2 *Submitted:* 2021-07-06
doi:10.18637/jss.v106.i02 *Accepted:* 2022-11-05