

# Package ‘viscomp’

January 16, 2023

**Type** Package

**Title** Visualize Multi-Component Interventions in Network Meta-Analysis

**Version** 1.0.0

**Maintainer** Georgios Seitidis <g.seitidis@uoi.gr>

**Description** A set of functions providing several visualization tools for exploring the behavior of the components in a network meta-analysis of multi-component (complex) interventions:

- components descriptive analysis
- heat plot of the two-by-two component combinations
- leaving one component combination out scatter plot
- violin plot for specific component combinations' effects
- density plot for components' effects
- waterfall plot for the interventions' effects that differ by a certain component combination
- network graph of components
- rank heat plot of components for multiple outcomes.

The implemented tools are described by Seitidis et al. (2023) <doi:10.1002/jrsm.1617>.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 2.10),

**Imports** circlize (>= 0.4.15), dplyr (>= 1.0.9), ggExtra (>= 0.10.0), ggnewscale (>= 0.4.8), ggplot2 (>= 3.3.6), Hmisc (>= 4.7.0), MASS (>= 7.3.56), netmeta (>= 1.3-0), plyr (>= 1.8.7), qgraph (>= 1.9.2), reshape2 (>= 1.4.4), tibble (>= 3.1.7), tidyr (>= 1.2.0)

**RoxygenNote** 7.2.0

**Suggests** covr, knitr, rmarkdown

**URL** <https://github.com/georgiosseitidis/viscomp>,  
<https://georgiosseitidis.github.io/viscomp/>

**BugReports** <https://github.com/georgiosseitidis/viscomp/issues>

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Georgios Seitidis [aut, cre] (<<https://orcid.org/0000-0003-0856-1892>>),  
 Sofia Tsokani [aut] (<<https://orcid.org/0000-0002-6258-8861>>),  
 Christos Christogiannis [aut],  
 Katerina Maria Kontouli [aut],  
 Alexandros Fyraridis [aut],  
 Stavros Nikolakopoulos [aut] (<<https://orcid.org/0000-0002-9769-3725>>),  
 Areti Angeliki Veroniki [aut] (<<https://orcid.org/0000-0001-6388-4825>>),  
 Dimitris Mavridis [aut] (<<https://orcid.org/0000-0003-1041-4592>>)

**Repository** CRAN

**Date/Publication** 2023-01-16 09:50:02 UTC

## R topics documented:

compdesc	2
compGraph	4
denscomp	5
heatcomp	6
loccos	8
MACE	9
nmaMACE	10
rankheatplot	10
specc	12
watercomp	14

**Index** **16**

---

compdesc	<i>Components descriptive analysis</i>
----------	--

---

## Description

The function performs a descriptive analysis regarding the frequency of the components in the network meta-analysis model.

## Usage

```
compdesc(model, sep = "+", heatmap = TRUE, percentage = TRUE, digits = 2)
```

## Arguments

model	An object of class <code>netmeta</code> .
sep	A single character that defines the separator between interventions components.
heatmap	logical. If TRUE a heat matrix of the component's frequency is plotted.
percentage	logical. If TRUE combinations' percentages are printed as a number instead of fraction value in the heatmap.
digits	A single integer value that specifies the percentages' decimal places in the heatmap.

**Value**

A list containing three items

crosstable	A cross-table containing the frequency of the components. Each cell represents the number of arms where the corresponding component combination was observed.
frequency	A data.frame that contains the component's frequency. Columns <ul style="list-style-type: none"> <li>• Component denotes the name of each component</li> <li>• Frequency denotes the number of arms where the component was observed</li> <li>• A denotes the number of studies in which the corresponding component was included in all arms</li> <li>• A_percent denotes the percentage of studies in which the corresponding component was included in all arms</li> <li>• B denotes the number of studies in which the corresponding component was included in at least one arm</li> <li>• B_percent denotes the percentage of studies in which the corresponding component was included in at least one arm</li> <li>• C denotes the number of studies in which the corresponding component was not included in any arm</li> <li>• C_percent denotes the percentage of studies in which the corresponding component was not included in any arm</li> <li>• A.B denotes the ratio of columns A and B.</li> </ul>
heatmat	An object of class <code>ggplot</code> that visualizes item <code>crosstable</code> . Diagonal elements refer to the components and in parentheses the proportion of study arms including that component is provided, while off-diagonal elements to the frequency of component's combinations and in parentheses the proportion of study arms with both components out of those study arms that have the component in the row is provided. Also, the intensity of the color is proportional to the relative frequency of the component combination.

**Note**

The function can be applied only in network meta-analysis models that contain multi-component interventions.

**Examples**

```
data(nmaMACE)
compdesc(model = nmaMACE)
```

---

`compGraph`*Components Network Graph*

---

### Description

The Components Network Graph is meant to visualize the frequency of components' combinations found in the network.

### Usage

```
compGraph(  
  model,  
  sep = "+",  
  mostF = 5,  
  excl = NULL,  
  title = "Most frequent combinations of components",  
  print_legend = TRUE,  
  size_legend = 0.825  
)
```

### Arguments

<code>model</code>	An object of class <code>netmeta</code> .
<code>sep</code>	A single character that defines the separator between interventions components.
<code>mostF</code>	Number of most frequent combinations of the network.
<code>excl</code>	A character vector that specifies the combinations to be excluded from the plot.
<code>title</code>	A single character that specifies the overall title of the plot.
<code>print_legend</code>	logical. If TRUE the legend is printed.
<code>size_legend</code>	size of the legend.

### Details

The function resembles a network plot where nodes represent the individual components found in the network and edges represent the combination of components found in at least one treatment arm of the trials included in the network meta-analysis model. Each edge's color represents one of the unique interventions (components' combination) found in the network of interventions. Edges' thickness indicates the frequency by which each intervention (combination of components) was observed in the network (number of arms in which the combination was assigned). The number of the most frequent combinations can be modified from the argument `mostF`. The function by default plots the five most frequent components' combinations found in the network.

### Value

Returns (invisibly) a `qgraph` object.

**Note**

The function can be applied only in network meta-analysis models that contain multi-component interventions.

**Examples**

```
data(nmaMACE)
compGraph(model = nmaMACE)
```

---

denscomp

*Components Density Plot*

---

**Description**

The function creates density plots in order to explore the efficacy of the components.

**Usage**

```
denscomp(  
  model,  
  sep = "+",  
  combination,  
  violin = FALSE,  
  random = TRUE,  
  z_value = FALSE  
)
```

**Arguments**

model	An object of class <code>netmeta</code> .
sep	A single character that defines the separator between interventions components.
combination	A character vector that contains the component combinations of interest.
violin	logical. If TRUE the density is visualized via violins instead of density plots.
random	logical. If TRUE the random-effects NMA model is used, instead of the fixed-effect NMA model.
z_value	logical. If TRUE z-values are used, instead intervention effects.

**Details**

If the length of the argument `combination` is 1, the function creates two density plots. The first is produced based on the interventions that include the component combination of interest (which is specified by the argument `combination`), while the second on the interventions that do not include the underlying component combination.

If the argument `combination` includes more than one elements, the number of densities is equal with the length of the argument `combination`, and each density is based on the interventions that

include the relative component combination. For example, if `combination = c("A + B", "B + C", "A")` the function will produce 3 density plots that are based on the interventions that includes components "A" and "B", the interventions that include components "B" and "C" and interventions that includes component "A", respectively.

The function by default uses the intervention's relative effects (`z_value = FALSE`) obtained from the random-effects network meta-analysis (NMA) model (`random = TRUE`). It can be also adjusted to use the intervention's z-values instead of the relative effects, by setting `z_value = TRUE`.

### Value

An object of class `ggplot`.

### Note

The efficacy of the components could be explored via violins plots instead of density plots, by setting `violin = TRUE`. Also, in the case of dichotomous outcomes, the log-scale is used.

The function can be applied only in network meta-analysis models that contain multi-component interventions.

### Examples

```
data(nmaMACE)
denscomp(model = nmaMACE, combination = "C")
```

---

heatcomp

*Components Heat Plot*

---

### Description

The function creates a heat plot based on the two-by-two component combinations, obtained from the network meta-analysis (NMA) model.

### Usage

```
heatcomp(
  model,
  sep = "+",
  median = TRUE,
  random = TRUE,
  z_value = FALSE,
  freq = TRUE,
  legend_name = NULL
)
```

## Arguments

<code>model</code>	An object of class <code>netmeta</code> .
<code>sep</code>	A single character that defines the separator between interventions components.
<code>median</code>	logical. If TRUE the median is used instead of the mean as a summary measure.
<code>random</code>	logical. If TRUE the random-effects NMA model is used instead of the fixed-effect NMA model.
<code>z_value</code>	logical. If TRUE z-values are used instead of interventions effects.
<code>freq</code>	logical. If TRUE the frequency of component combinations are printed.
<code>legend_name</code>	A single character that specifies the title of the legend.

## Details

Diagonal elements refer to components, while off-diagonal to components' combinations. Each element summarizes by default the NMA relative effects (`z_value = FALSE`) of the interventions that includes the corresponding component combination. Combinations that were not observed in the NMA model, are denoted by the letter "X". Frequency of component combinations observed in the NMA is printed by default (`freq = TRUE`). As a summary measure, the median is used by default (`median = TRUE`). The magnitude of each relative effect is reflected by the color's intensity. Estimates close to zero are denoted by white color, and indicates a small magnitude of the corresponding component combination, while deep green and red colors indicate a large magnitude of the corresponding component combination. Outcomes nature (beneficial or harmful) is defined in the `netmeta` model.

The function can be also adjusted to include z-scores by setting the argument `z_value = TRUE`. Z-scores quantify the strength of statistical evidence. Thus, dark green (or red) indicates strong statistical evidence that the corresponding component (or combination of components) performs better (or worse) than the reference intervention.

## Value

An object of class `ggplot`.

## Note

In the case where the NMA relative effects are used, the uncertainty of the NMA estimates are reflected by the size of the grey boxes. The bigger the box, the more precise the estimate.

By setting `median = FALSE`, the mean is used instead of the median as a summary measure.

The function can be applied only in network meta-analysis models that contain multi-component interventions.

## Examples

```
data(nmaMACE)
heatcomp(model = nmaMACE)
```

**Description**

The function based on the network meta-analysis (NMA) estimates explores if a set of components has a positive or a negative impact on the outcome, by creating a scatter plot based on the set of interventions that differ by a specific set of components.

**Usage**

```
Ioccos(
  model,
  sep = "+",
  combination = NULL,
  random = TRUE,
  z_value = FALSE,
  histogram = TRUE,
  histogram.color = "blue"
)
```

**Arguments**

<code>model</code>	An object of class <code>netmeta</code> .
<code>sep</code>	A single character that defines the separator between interventions components.
<code>combination</code>	A single character that specifies the component combination of interest.
<code>random</code>	logical. If TRUE the random-effects NMA model is used instead of the fixed-effect NMA model.
<code>z_value</code>	logical. If TRUE z-values are used instead of interventions effects.
<code>histogram</code>	logical. If TRUE histograms are added to the plot.
<code>histogram.color</code>	A single character that specifies the color of the histogram. See <code>ggMarginal</code> for more details.

**Details**

Axis y represents the intervention's effect when the component combination is not included in the intervention, while axis x represents the intervention's effect when is included. Line  $y = x$  splits the plot in two parts. For a beneficial outcome, dots above the line indicates that the inclusion of component combination balk the intervention's efficacy, while dots below the line indicate that the inclusion of the component combination increases intervention's efficacy. The opposite holds for harmful outcomes.

The component combination of interest is specified by the argument `combination`. For example, if `combination = "A"`, the function plots all the interventions that differ by the component "A". If `combination = NULL`, all interventions that differ by one component are plotted.



The function by default uses the NMA relative effects estimates, but it can be adjusted to use the z-values by setting the argument `z_value = TRUE`. Histograms for the nodes that include and not include the component combination can be added to the scatter plot, by setting the argument `histogram = TRUE`.

### Value

An object of class `ggplot`.

### Note

In the case of dichotomous outcomes, the log-scale is used for both axis. Also, the function can be applied only in network meta-analysis models that contain multi-component interventions.

### Examples

```
data(nmaMACE)
loccos(model = nmaMACE, combination = c("B"))
```

---

MACE

*Major Adverse Cardiovascular Event*

---

### Description

An artificial network meta-analysis data comparing the effectiveness of a number of interventions for major adverse cardiovascular events.

### Usage

```
MACE
```

### Format

A `data.frame` with the following columns

**Study** The study name of the trial

**treat1, treat2, treat3, treat4** Treatment names of arms

**n1, n2, n3, n4** Total number of participants in arms

**event1, event2, event3, event4** Total number of events in arms

---

 nmaMACE

*Network Meta-Analysis of Major Adverse Cardiovascular Event*


---

### Description

An artificial network meta-analysis (of class `netmeta`) comparing the effectiveness of a number of interventions for major adverse cardiovascular events.

### Usage

```
nmaMACE
```

### Format

An object of class `netmeta` of length 157.

---

rankheatplot

*Components Rank Heat Plot*


---

### Description

Rank heat plot summarizes the components' p-scores for multiple outcomes.

### Usage

```
rankheatplot(
  model,
  sep = "+",
  median = TRUE,
  random = TRUE,
  outcomeNames = NULL,
  cex_components = NULL,
  cex_values = NULL,
  cex_outcomes = NULL
)
```

### Arguments

<code>model</code>	A list of <code>netmeta</code> models.
<code>sep</code>	A single character that defines the separator between interventions components.
<code>median</code>	logical. If TRUE the median is used as a summary measure instead of the mean.
<code>random</code>	A logical vector that specifies the NMA model for each outcome. If TRUE the random-effects NMA model is used instead of the fixed-effects NMA model.
<code>outcomeNames</code>	A character vector that specifies the names of the outcomes.

cex\_components Font size of components' names.  
 cex\_values Font size of p-scores.  
 cex\_outcomes Font size of outcomes' names.

### Details

The function creates a rank heat plot, where the number of circles depend on the number of outcomes. Each circle is divided by the total number of components, and each sector is colored according to the corresponding component p-score. Components' p-scores are summarized by using either the median (median = TRUE) or the mean (median = FALSE) of the p-scores obtained from the interventions that include the corresponding component. The sector's colors reflect the magnitude of the components p-scores. Red color indicates a low p-score (close to zero), while green color indicates values close to 1. Intervention's p-scores are obtained from the network meta-analysis (NMA) model. By default the random-effects NMA model is used for each outcome (random = TRUE).

### Value

Returns (invisibly) a rank heat plot.

### Note

The function can be applied only in network meta-analysis models that contain multi-component interventions.

### Examples

```
# Artificial data set

t1 <- c("A", "B", "C", "A+B", "A+C", "B+C", "A")
t2 <- c("C", "A", "A+C", "B+C", "A", "B", "B+C")

TE1 <- c(2.12, 3.24, 5.65, -0.60, 0.13, 0.66, 3.28)
TE2 <- c(4.69, 2.67, 2.73, -3.41, 1.79, 2.93, 2.51)

seTE1 <- rep(0.1, 7)
seTE2 <- rep(0.2, 7)

study <- paste0("study_", 1:7)

data1 <- data.frame(
  "TE" = TE1, "seTE" = seTE1, "treat1" = t1, "treat2" = t2, "studlab" = study,
  stringsAsFactors = FALSE
)

data2 <- data.frame(
  "TE" = TE2, "seTE" = seTE2, "treat1" = t1, "treat2" = t2, "studlab" = study,
  stringsAsFactors = FALSE
)
```

```

# Network meta-analysis models

net1 <- netmeta::netmeta(
  TE = TE, seTE = seTE, studlab = studlab, treat1 = treat1,
  treat2 = treat2, data = data1, ref = "A"
)

net2 <- netmeta::netmeta(
  TE = TE, seTE = seTE, studlab = studlab, treat1 = treat1,
  treat2 = treat2, data = data2, ref = "A"
)

# Rank heat plot

rankheatplot(model = list(net1, net2))

```

---

specc

*Specific Component Combination violin plots*


---

### Description

The function based on the network meta-analysis (NMA) estimates produces violin plots from interventions that include the component combinations of interest.

### Usage

```

specc(
  model,
  sep = "+",
  combination = NULL,
  components_number = FALSE,
  groups = NULL,
  random = TRUE,
  z_value = FALSE,
  prop_size = TRUE,
  fill_violin = "lightblue",
  color_violin = "lightblue",
  adj_violin = 1,
  width_violin = 1,
  boxplot = TRUE,
  width_boxplot = 0.5,
  errorbar_type = 5,
  dots = TRUE,
  jitter_shape = 16,
  jitter_position = 0.01,
  values = TRUE
)

```

## Arguments

<code>model</code>	An object of class <a href="#">netmeta</a> .
<code>sep</code>	A single character that defines the separator between interventions components.
<code>combination</code>	A character vector that specifies the component combinations of interest.
<code>components_number</code>	logical. If TRUE the violins are created based on the number of components included in the interventions.
<code>groups</code>	A character vector that contains the clusters of the number of components. Elements of the vector must be integer numbers (e.g. 5 or "5"), or range values (e.g. "3-4" ), or in the "xx+" format (e.g "5+").
<code>random</code>	logical. If TRUE the random-effects NMA model is used instead of the fixed-effect NMA model.
<code>z_value</code>	logical. If TRUE z-values are used instead of interventions effects.
<code>prop_size</code>	logical. If TRUE in the case where <code>z_value == FALSE</code> , the size of the dots is proportional to the precision of the estimates.
<code>fill_violin</code>	fill color of the violin. See <a href="#">geom_violin</a> for more details.
<code>color_violin</code>	color of the violin. See <a href="#">geom_violin</a> for more details.
<code>adj_violin</code>	adjustment of the violin. See <a href="#">geom_violin</a> for more details.
<code>width_violin</code>	width of the violin. See <a href="#">geom_violin</a> for more details.
<code>boxplot</code>	logical. If TRUE boxplots are plotted.
<code>width_boxplot</code>	width of the boxplot. See <a href="#">geom_boxplot</a> for more details.
<code>errorbar_type</code>	boxplot's line type. See <a href="#">stat_boxplot</a> for more details.
<code>dots</code>	logical. If TRUE data points are plotted.
<code>jitter_shape</code>	jitter shape. See <a href="#">geom_jitter</a> for more details.
<code>jitter_position</code>	jitter position. See <a href="#">geom_jitter</a> for more details.
<code>values</code>	logical. If TRUE median value of each violin is printed.

## Details

By default the function creates a violin for each component of the network (`combination = NULL`). Each violin visualizes the distribution of the effect estimates, obtained from the interventions that include the corresponding component. Combinations of interest are specified from the argument `combination`. For example, if `combination = c("A", "A + B")`, two violin plots are produced. The first one is based on the interventions that contain the component "A", and the second one, based on the interventions that contain both components A and B.

By setting the argument `components_number = TRUE`, the behavior of intervention's effect as the number of components increased is explored, by producing violins based on the number of components included in the interventions. If the number of components included in a intervention ranges between 1 and 3, then 3 violins will be produced in total. The violins will be based on the interventions that include one component, two components, and three components respectively. The number of components could be also categorized in groups by the argument `groups`. For example

if `components_number = TRUE` and `groups = c("1-3", 4, "5+")`, 3 violins will be created. One for the interventions that contain less than 3 components, one for the interventions that contain 4 components and one for those that contain more than 5 components.

The function by default uses the NMA relative effects, but it could be adjusted to use intervention's z-scores by setting `z_value = TRUE`. In the case where the NMA relative effects, the size of dots reflects the precision of the estimates. Larger dots indicates more precise NMA estimates.

### Value

An object of class `ggplot`.

### Note

In the case of dichotomous outcomes, the log-scale is used in axis y. Also, the function can be applied only in network meta-analysis models that contain multi-component interventions.

### Examples

```
data(nmaMACE)
specc(model = nmaMACE, combination = c("B", "C", "B + C"))
```

---

watercomp

*Waterfall plot*

---

### Description

The function produces a waterfall plot based on the z-values from the interventions that differ by one specific component combination.

### Usage

```
watercomp(model, sep = "+", combination = NULL, z_value = FALSE, random = TRUE)
```

### Arguments

<code>model</code>	An object of class <code>netmeta</code> .
<code>sep</code>	A single character that defines the separator between interventions components.
<code>combination</code>	A single character that specifies the component combination of interest.
<code>z_value</code>	logical. If TRUE z-values are used instead of interventions effects.
<code>random</code>	logical. If TRUE z-values are obtained from the random-effects NMA model instead of the fixed-effect NMA model.

**Details**

The function based on the intervention's z-values (default choice) obtained from the network meta-analysis (NMA) model visualizes all the observed interventions that differ by one specific component combination, in order to explore if the one extra component combination from every comparison has a positive or negative impact. Bars above or below of the  $y = 0$  line, indicates that the inclusion of the extra specific component combination has an impact on the intervention. The direction of the impact (positive or negative), depends on the outcomes' nature (beneficial or harmful).

The combination of interest is defined from the argument `combination`. By default the function visualizes the interventions that differ by one component (`combination = NULL`). If for example `combination = "A+B"`, the function plots the interventions that differ by "A+B".

**Value**

An object of class `ggplot`.

**Note**

In the case of dichotomous outcomes, the log-scale is used in axis y. Also, the function can be applied only in network meta-analysis models that contain multi-component interventions.

**Examples**

```
data(nmaMACE)
watercomp(nmaMACE)
```

# Index

## \* datasets

MACE, [9](#)

nmaMACE, [10](#)

compdesc, [2](#)

compGraph, [4](#)

denscomp, [5](#)

geom\_boxplot, [13](#)

geom\_jitter, [13](#)

geom\_violin, [13](#)

ggMarginal, [8](#)

ggplot, [9](#)

heatcomp, [6](#)

loccos, [8](#)

MACE, [9](#)

netmeta, [2](#), [4](#), [5](#), [7](#), [8](#), [10](#), [13](#), [14](#)

nmaMACE, [10](#)

qgraph, [4](#)

rankheatplot, [10](#)

specc, [12](#)

stat\_boxplot, [13](#)

watercomp, [14](#)