Package 'rankinma'

April 1, 2024

Type Package

Title Rank in Network Meta-Analysis

Version 0.2.2

Date 2024-04-01

Maintainer Enoch Kang <y.enoch.kang@gmail.com>

Description A supportive collection of functions for gathering and plotting treatment ranking metrics after network meta-analysis.

License GPL (>= 3)

URL <https://rankinma.shinyapps.io/rankinma/>

Depends R (>= 4.2.0)

Imports graphics, grDevices, mvtnorm, netmeta, stats, utils

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Language en-US

Encoding UTF-8

RoxygenNote 7.2.2

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Chiehfeng Chen [aut] (<https://orcid.org/0000-0002-1595-6553>),
Enoch Kang [aut, cre, trl] (<https://orcid.org/0000-0002-4903-942X>),
Wen-Hsuan Hou [aut] (<https://orcid.org/0000-0002-4376-6298>),
Jin-Hua Chen [aut] (<https://orcid.org/0000-0002-3130-4125>),
Yu-Chieh Chuang [aut] (<https://orcid.org/0000-0002-7124-6556>),
Edwin Chan [aut]

Repository CRAN

Date/Publication 2024-04-01 13:50:02 UTC

R topics documented:

---

rankinma-package   *rankinma: Rank in Network Meta-Analysis*

Description

rankinma is an R package that supports users to easily obtain and visualize various metrics of treatment ranking from network meta-analysis no matter using either frequentist or Bayesian approach. Development of package *rankinma* is based on **R version 4.2.2 (2022-10-31 ucrt)**. Extra imported packages are as follows:

- *mvtnorm* (version 1.1-3)
- *netmeta* (version 2.6-0)

Details

Current version consists of seven functions, including two functions for data preparation (function `GetMetrics` and `SetMetrics`) and five functions for visualization of treatment ranking metrics (i.e. `PlotBeads`, `PlotLine`, `PlotBar`, `PlotHeat`, and `PlotSpie`). Probabilities of treatments on each possible rank can be visualized using `PlotLine` and `PlotBar`. Due to concise information, `PlotBeads` is recommended to be used for global metrics of treatment ranking, such as **P-score** and **SUCRA**. The other four visualization functions can also generate graphics of the global metrics.

References

1. Salanti, G., Ades, A. E., & Ioannidis, J. P. (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.

2. Chaimani, A., Higgins, J. P., Mavridis, D., Spyridonos, P., & Salanti, G. (2013). Graphical tools for network meta-analysis in STATA. **PloS one**, *8(10)*, e76654.

3. Van Valkenhoef, G., Tervonen, T., Zwinkels, T., De Brock, B., & Hillege, H. (2013). ADDIS: a decision support system for evidence-based medicine. **Decision Support Systems**, *55(2)*, 459-475.

4. Rücker, G., & Schwarzer, G. (2015). Ranking treatments in frequentist network meta-analysis works without resampling methods. **BMC medical research methodology**, *15(1)*, 1-9.

5. Daly, C. H., Mbuagbaw, L., Thabane, L., Straus, S. E., & Hamid, J. S. (2020). Spie charts for quantifying treatment effectiveness and safety in multiple outcome network meta-analysis: a proof-of-concept study. **BMC Medical Research Methodology**, *20*, 1-13.

6. Balduzzi, S., Rücker, G., Nikolakopoulou, A., Papakonstantinou, T., Salanti, G., Efthimiou, O., & Schwarzer, G. (2023). netmeta: An R package for network meta-analysis using frequentist methods. **Journal of Statistical Software**, *106*, 1-40.

GetMetrics	*Get treatment ranking metrics from network meta-analysis output*

Description

GetMetrics() is a function for gathering metrics of treatment ranking from *netmeta* output.

Usage

```
GetMetrics(
  data,
  outcome = NULL,
  prefer = NULL,
  metrics = NULL,
  model = "random",
  simt = 1000,
  rob = NULL
)
```

Arguments

data	DATA of *netmeta* output.
outcome	STRING for name of outcome.
prefer	STRING for indicating which direction is beneficial treatment effect in terms of "small" and "large" values in statistic test.
metrics	STRING for metrics of treatment ranking in terms of "SUCRA", "P-score", and "P-best" for the value of surface under the cumulative ranking curve, P-score, and probability of achieving the best treatment.
model	STRING for analysis model in terms of "random" and "common" for random-effects model and common-effect model.
simt	INTEGER for times of simulations to estimate surface under the cumulative ranking curve (SUCRA).
rob	STRING for column name of risk of bias.

Value

GetMetrics() returns a data.frame with three columns, including treatment, metrics of treatment ranking, and outcome name.

References

Rücker, G., & Schwarzer, G. (2015). Ranking treatments in frequentist network meta-analysis works without resampling methods. **BMC medical research methodology**, *15(1)*, 1-9.

Salanti, G., Ades, A. E., & Ioannidis, J. P. (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.

See Also

SetMetrics

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#dataMetrics <- GetMetrics(nma, outcome = "HbA1c", prefer = "small",
#metrics = "SUCRA", model = "random", simt = 1000)

# Get P-score
#dataMetrics <- GetMetrics(nma, outcome = "HbA1c", prefer = "small",
#metrics = "P-score", model = "random", simt = 1000)
## End(Not run)
```

| PlotBar | *Illustrate bar chart of treatment ranking metrics* |

Description

PlotBar() is a function for illustrating bar chart in both separated and accumulative styles.

Usage

```
PlotBar(data, accum = NULL, merge = NULL, color = NULL, rotateX = NULL)
```

Arguments

data	DATA of metrics for treatment ranking.
accum	LOGIC value for indicating whether use accumulative probabilities. This parameter is only for probabilities but not global metrics of treatment ranking.
merge	LOGIC value for indicating whether merge bar charts together.
color	LIST of colors for treatments in a network meta-analysis, or CHARACTER of a color for the bar on not accumulated bar chart.
rotateX	NUMERIC value between 0 and 360 for rotating x axis labels of bars.

Value

PlotBar() returns a bar chart.

References

Van Valkenhoef, G., Tervonen, T., Zwinkels, T., De Brock, B., & Hillege, H. (2013). ADDIS: a decision support system for evidence-based medicine. **Decision Support Systems**, *55(2)*, 459-475.

See Also

GetMetrics, SetMetrics, PlotBeads, PlotLine, PlotHeat, PlotSpie

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#dataMetrics <- GetMetrics(nma, outcome = "HbA1c", prefer = "small", metrics = "SUCRA",
#model = "random", simt = 1000)

# Set data for rankinma
#dataRankinma <- SetMetrics(dataMetrics, tx = tx, outcome = outcome,
#metrics = SUCRA, metrics.name = "SUCRA")

# Illustrate bar plot
#PlotBar(dataRankinma)
## End(Not run)
```

PlotBeads   *Illustrate beading plot*

Description

PlotBeads() is a function for illustrating beading plot.

Usage

```
PlotBeads(
  data,
  scaleX = "Numeric",
  txtValue = "Effects",
  color = NULL,
  whichRoB = "None",
```

```
    lgcBlind = FALSE,
    szPnt = NULL,
    szFntTtl = NULL,
    szFntTtlX = NULL,
    szFntX = NULL,
    szFntY = NULL,
    szFntTxt = NULL,
    szFntLgnd = NULL,
    rotateTxt = 60
)
```

Arguments

data	DATA of metrics for treatment ranking.
scaleX	STRING for indicating scale on the x axis.
txtValue	STRING for indicating labels of metrics or effects on each point.
color	LIST of colors for treatments in a network meta-analysis.
whichRoB	STRING for indicating how to display risk of bias for each treatment.
lgcBlind	LOGIC value for indicating whether to display with color-blind friendly.
szPnt	NUMERIC value for indicating point size of ranking metrics.
szFntTtl	NUMERIC value for indicating font size of main title.
szFntTtlX	NUMERIC value for indicating font size of title on X-axis.
szFntX	NUMERIC value for indicating font size of numeric scale on X-axis.
szFntY	NUMERIC value for indicating font size of outcome name(s).
szFntTxt	NUMERIC value for indicating font size of value of each point.
szFntLgnd	NUMERIC value for indicating legend font size.
rotateTxt	NUMERIC value between 0 and 360 for rotating labels of text values of each point.

Value

PlotBeads() returns a beading plot.

Author(s)

Chiehfeng Chen & Enoch Kang

References

Chen, C., Chuang, Y.C., Chan, E., Chen, J.H., Hou, W.H., & Kang, E. (2023). Beading plot: A novel graphics for ranking interventions in network evidence. PREPRINT (Version 1) available at Research Square.

See Also

[GetMetrics](), [SetMetrics]()

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#nma.1 <- GetMetrics(nma, outcome = "HbA1c.random", prefer = "small", metrics = "SUCRA",
#model = "random", simt = 1000)
#nma.2 <- GetMetrics(nma, outcome = "HbA1c.common", prefer = "small", metrics = "SUCRA",
#model = "common", simt = 1000)

# Combine metrics of multiple outcomes
#dataMetrics <- rbind(nma.1, nma.2)

# Set data for rankinma
#dataRankinma <- SetMetrics(dataMetrics, tx = tx, outcome = outcome,
#metrics = SUCRA, metrics.name = "SUCRA")

# Illustrate beading plot
#PlotBeads(data = dataRankinma)
## End(Not run)
```

PlotHeat   *Illustrate heat plot for treatment ranking*

Description

PlotHeat() is a function for illustrating heat plot.

Usage

```
PlotHeat(data, sorttx = NULL, rotateX = NULL, szFntY = NULL)
```

Arguments

data	DATA of metrics for treatment ranking.
sorttx	LOGIC value for indicating whether sort heat plot by treatments.
rotateX	NUMERIC value between 0 and 360 for rotating x axis labels of heat plot.
szFntY	NUMERIC value for indicating font size of outcome name(s).

Value

PlotHeat() returns a heat plot.

See Also

[GetMetrics](#), [SetMetrics](#), [PlotBeads](#), [PlotBar](#), [PlotLine](#), [PlotSpie](#)

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#nma.1 <- GetMetrics(nma, outcome = "HbA1c.random", prefer = "small", metrics = "SUCRA",
#model = "random", simt = 1000)
#nma.2 <- GetMetrics(nma, outcome = "HbA1c.common", prefer = "small", metrics = "SUCRA",
#model = "common", simt = 1000)

# Combine metrics of multiple outcomes
#dataMetrics <- rbind(nma.1, nma.2)

# Set data for rankinma
#dataRankinma <- SetMetrics(dataMetrics, tx = tx, outcome = outcome,
#metrics = SUCRA, metrics.name = "SUCRA")

# Illustrate heat plot
#PlotHeat(data = dataRankinma)
## End(Not run)
```

PlotLine   *Illustrate line chart of treatment ranking metrics*

Description

PlotLine() is a function for illustrating line chart in both simple and composite styles.

Usage

```
PlotLine(
  data,
  accum = NULL,
  compo = NULL,
  merge = NULL,
  color = NULL,
  rotateX = NULL
)
```

Arguments

`data`	DATA of metrics for treatment ranking.
`accum`	LOGIC value for indicating whether use accumulative probabilities. This parameter is only for probabilities but not global metrics of treatment ranking.
`compo`	LOGIC value for indicating whether use composite line chart. This parameter is only for probabilities but not global metrics of treatment ranking.
`merge`	LOGIC value for indicating whether merge line charts together.
`color`	LIST of colors for treatments in a network meta-analysis, or CHARACTER of a color for the line on not composite line chart.
`rotateX`	NUMERIC value between 0 and 360 for rotating x axis labels of line chart.

Value

PlotLine() returns a line chart.

References

Chaimani, A., Higgins, J. P., Mavridis, D., Spyridonos, P., & Salanti, G. (2013). Graphical tools for network meta-analysis in STATA. **PloS one**, *8(10)*, e76654.

See Also

[GetMetrics](#), [SetMetrics](#), [PlotBeads](#), [PlotBar](#), [PlotHeat](#), [PlotSpie](#)

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#dataMetrics <- GetMetrics(nma, outcome = "HbA1c", prefer = "small", metrics = "SUCRA",
#model = "random", simt = 1000)

# Set data for rankinma
#dataRankinma <- SetMetrics(dataMetrics, tx = tx, outcome = outcome,
#metrics = SUCRA, metrics.name = "SUCRA")

# Illustrate bar plot
#PlotLine(dataRankinma)
## End(Not run)
```

PlotSpie   *Illustrate beading plot*

Description

PlotSpie() is a function for illustrating spie plot.

Usage

```
PlotSpie(data, color = NULL)
```

Arguments

data	DATA of metrics for treatment ranking.
color	LIST of colors for outcomes in a network meta-analysis.

Value

PlotSpie() returns a spie plot.

References

Daly, C. H., Mbuagbaw, L., Thabane, L., Straus, S. E., & Hamid, J. S. (2020). Spie charts for quantifying treatment effectiveness and safety in multiple outcome network meta-analysis: a proof-of-concept study. **BMC Medical Research Methodology**, *20*, 1-13.

See Also

[GetMetrics](), [SetMetrics](), [PlotBeads](), [PlotBar](), [PlotLine](), [PlotHeat]()

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#nma.1 <- GetMetrics(nma, outcome = "HbA1c.random", prefer = "small", metrics = "SUCRA",
#model = "random", simt = 1000)
#nma.2 <- GetMetrics(nma, outcome = "HbA1c.common", prefer = "small", metrics = "SUCRA",
#model = "common", simt = 1000)

# Combine metrics of multiple outcomes
#dataMetrics <- rbind(nma.1, nma.2)

# Set data for rankinma
#dataRankinma <- SetMetrics(dataMetrics, tx = tx, outcome = outcome,
```

```
#metrics = SUCRA, metrics.name = "SUCRA")

# Illustrate beading plot
#PlotSpie(data = dataRankinma)
## End(Not run)
```

SetMetrics   *Setup data of treatment ranking metrics for rankinma*

Description

SetMetrics() is a function for checking and preparing data set of metrics for further ploting in
rankinma.

Usage

```
SetMetrics(
  data,
  outcome = NULL,
  tx = NULL,
  metrics = NULL,
  metrics.name = NULL,
  trans = 0.8
)
```

Arguments

`data`	DATAFRAME of treatment, metrics, and name of outcomes.
`outcome`	VARIABLE string data for of outcome(s).
`tx`	VARIABLE with string data for treatments.
`metrics`	VARIABLE with numeric data for global metrics, but it should be "NULL" when using "Probabilities" as metrics.
`metrics.name`	STRING for metrics of treatment ranking in terms of "SUCRA","P-score", and "P-best" for the value of surface under the cumulative ranking curve, P-score, and probability of achieving the best treatment.
`trans`	NUMERIC for indicating transparency of colors of treatments.

Value

SetMetrics() returns a confirmed data.frame of treatment, metrics of treatment ranking, and outcome name.

`metrics.name`	A string shows type of metrics of treatment ranking.
`ls.outcome`	Strings list outcomes.
`ls.tx`	Strings list treatments.

`n.outcome`	An integer shows numbers of outcomes.
`n.tx`	An integer shows numbers of treatments.
`data`	A data frame consists of seven columns of core information among all outcomes.
`data.sets`	A list shows data frame of core information by each outcome.
`ptrn.tx`	A data frame shows treatments on each outcome.
`ptrn.outcome`	A data frame shows outcomes by treatments.
`color.txs`	A data frame shows color of each treatment.
`trans`	A numeric value shows transparency for colors of each treatment.

See Also

[GetMetrics](#)

Examples

```
## Not run:
#library(netmeta)
#data(Senn2013)
#nma <- netmeta(TE, seTE, treat1, treat2,
#studlab, data = Senn2013, sm = "SMD")

# Get SUCRA
#nma.1 <- GetMetrics(nma, outcome = "HbA1c.random", prefer = "small", metrics = "SUCRA",
#model = "random", simt = 1000)
#nma.2 <- GetMetrics(nma, outcome = "HbA1c.common", prefer = "small", metrics = "SUCRA",
#model = "common", simt = 1000)

# Combine metrics of multiple outcomes
#dataMetrics <- rbind(nma.1, nma.2)

# Set data for rankinma
#dataRankinma <- SetMetrics(dataMetrics, tx = tx, outcome = outcome,
#metrics = SUCRA, metrics.name = "SUCRA")
## End(Not run)
```

Index