

Package: triangulate (via r-universe)

May 16, 2026

Title Useful functions for performing evidence triangulation

Version 0.0.2

Description What the package does (one paragraph).

License MIT + file LICENSE

VignetteBuilder knitr

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

URL <https://mcguinlu.github.io/triangulate>,
<https://github.com/mcguinlu/triangulate>,
<https://mrcieu.r-universe.dev/triangulate>

BugReports <https://github.com/mcguinlu/triangulate/issues>

Suggests testthat, knitr, readxl, rmarkdown, tibble

Config/testthat/edition 3

Imports dplyr, tidyr, stringr, magrittr, janitor, ggplot2, metadat,
metafor, purrr, scales, shiny

Depends R (>= 3.5)

Config/pak/sysreqs cmake make libicu-dev libuv1-dev zlib1g-dev

Repository <https://mrcieu.r-universe.dev>

Date/Publication 2026-01-16 16:25:15 UTC

RemoteUrl <https://github.com/mcguinlu/triangulate>

RemoteRef HEAD

RemoteSha 9c198ec995a81629a4cba46b38b577cbfa9b6a11

Contents

beta_carotene_data	2
dat_bias	3
dat_bias_values	3
dat_ind	3
dat_ind_values	4
interactive_bias_plot	4
tri_absolute_direction	5
tri_absolute_direction_invert	5
tri_absolute_direction_quick	6
tri_append_bias	6
tri_append_indirect	7
tri_calculate_adjusted_estimates	7
tri_dat_check	8
tri_plot_bias_direction	8
tri_prep_data	9
tri_swap_effect_direction	10
tri_to_long	10
tri_to_wide	11

Index	12
--------------	-----------

beta_carotene_data	<i>Beta-carotene RoB Judgments (Raw) Domain-level risk of bias judgments for 15 studies of beta-carotene and cardiovascular outcomes.</i>
--------------------	---

Description

Beta-carotene RoB Judgments (Raw) Domain-level risk of bias judgments for 15 studies of beta-carotene and cardiovascular outcomes.

Usage

```
beta_carotene_data
```

Format

A data frame with 27 rows and 25 variables.

Source

Supplement to Shapland et al. (2024)

dat_bias	<i>Example risk of bias assessments</i>
----------	---

Description

Example risk of bias assessments

Usage

dat_bias

Format

A data frame with 20 rows and 26 variables

dat_bias_values	<i>Example priors for the bias adjustment</i>
-----------------	---

Description

Example priors for the bias adjustment

Usage

dat_bias_values

Format

A data frame with 3 rows and 6 variables

dat_ind	<i>Example indirectness assessments</i>
---------	---

Description

Example indirectness assessments

Usage

dat_ind

Format

A data frame with 20 rows and 14 variables

dat_ind_values	<i>Example priors for indirectness adjustment</i>
----------------	---

Description

Example priors for indirectness adjustment

Usage

```
dat_ind_values
```

Format

A data frame with 3 rows and 6 variables

interactive_bias_plot	<i>Interactive bias-adjusted plot (horizontal layout)</i>
-----------------------	---

Description

Launches a Shiny app that allows users to adjust additive and proportional bias priors and visualizes both original and adjusted estimates horizontally.

Usage

```
interactive_bias_plot(dat)
```

Arguments

dat	A data.frame with columns: result_id, yi, vi, study, type
-----	---

Value

A Shiny app object

`tri_absolute_direction`*Convert to absolute direction of bias or indirectness*

Description

Converts qualitative bias direction labels (e.g. "Favours comparator") into absolute directions ("left" or "right") based on effect size and bias type.

Usage

```
tri_absolute_direction(dat)
```

Arguments

<code>dat</code>	Data in long format
------------------	---------------------

`tri_absolute_direction_invert`*Convert absolute direction back to qualitative tool labels*

Description

Reverses the transformation done by `tri_absolute_direction()`, converting absolute directions ("left"/"right") back to labels like "Favours comparator".

Usage

```
tri_absolute_direction_invert(dat)
```

Arguments

<code>dat</code>	Data in long format
------------------	---------------------

tri_absolute_direction_quick

Quick wrapper to apply absolute direction transformation

Description

Converts data between wide and long format while applying or reversing the absolute direction transformation.

Usage

```
tri_absolute_direction_quick(dat, invert = FALSE)
```

Arguments

dat	Dataset in wide format ## check this is the correct parameter
invert	If TRUE, reverse transformation (absolute → qualitative)

tri_append_bias

Append bias priors to bias dataset

Description

Joins the appropriate prior parameters to a long-format bias dataset, based on the risk-of-bias judgment (j), direction (d), and type (t).

Usage

```
tri_append_bias(dat, values = NULL, common = TRUE)
```

Arguments

dat	A bias dataset in long format (must include j, d, t, domain, result_id)
values	A data frame of prior distributions (e.g., from dat_bias_values)
common	Should a single set of prior distributions be used across all domains (default is TRUE)? Set to FALSE to allow domain-specific priors.

Value

Dataset with numeric bias prior values appended

tri_append_indirect *Append indirectness priors to dataset*

Description

Similar to tri_append_bias(), but operates on indirectness judgments and priors

Usage

```
tri_append_indirect(dat, values, common = TRUE)
```

Arguments

dat	A long-format indirectness dataset
values	A dataframe of indirectness priors (e.g., dat_ind_values)
common	Should a single set of distributions be used across all domains (default is TRUE)?

Value

Dataset with indirectness prior values appended

tri_calculate_adjusted_estimates
Calculate Bias-Adjusted Estimates

Description

Computes adjusted effect estimates (yi_adj) and variances (vi_adj) incorporating additive and proportional bias adjustments.

Usage

```
tri_calculate_adjusted_estimates(dat)
```

Arguments

dat	A dataset prepared using tri_prep_data(), containing bias parameters
-----	--

Value

A tibble with added yi_adj and vi_adj columns

tri_dat_check	<i>Check that required columns are present</i>
---------------	--

Description

Validates whether key variables are present in the dataset. Can toggle between a basic ("minimal") check or a "full" check including columns needed for bias adjustment.

Usage

```
tri_dat_check(dat, mode = c("minimal", "full"))
```

Arguments

dat	A data frame (long or wide format)
mode	Check mode: either "minimal" or "full"

Value

Throws an error if required columns are missing. Otherwise returns TRUE (invisibly).

tri_plot_bias_direction	<i>Plot bias-adjusted and unadjusted estimates with RoB</i>
-------------------------	---

Description

Creates a forest plot of both raw and adjusted effect estimates, alongside domain-level risk of bias annotations for each study.

Usage

```
tri_plot_bias_direction(  
  dat,  
  dat_adj = NULL,  
  title = NULL,  
  grouping = "type",  
  rma_method = "REML",  
  ...  
)
```

Arguments

dat	A dataframe output from <code>tri_calculate_adjusted_estimates()</code> , with <code>yi</code> , <code>vi</code> , <code>yi_adj</code> , <code>vi_adj</code> , and <code>domain</code> columns.
dat_adj	A data frame with adjusted estimates and variances, if different from the main input.
title	Optional plot title.
grouping	Column by which to stratify subgroups (default = "type").
rma_method	Meta-analysis method passed to <code>metafor::rma</code> (e.g., "REML", "FE"). We use <code>metafor</code> 's default which random-effects
...	Additional arguments passed to <code>rob_direction()</code> .

Value

A forest plot is drawn (base graphics).

tri_prep_data	<i>Prepare bias and indirectness data for adjustment</i>
---------------	--

Description

Aggregates domain-level prior data and calculates total adjustment values per study result. Supports additive and proportional adjustments.

Usage

```
tri_prep_data(dat_bias, dat_ind)
```

Arguments

dat_bias	Data with bias priors (long format)
dat_ind	Data with indirectness priors (long format)

Value

Data with `yi/vi` and adjustment components ready for adjustment

 tri_swap_effect_direction

Standardise Effect Direction and Adjust Bias Directions

Description

Flips effect estimates (y_i) and proportional bias directions (d) for specified study types (e.g., odds ratios) to ensure consistency across studies using different metrics.

Usage

```
tri_swap_effect_direction(dat, types = NULL)
```

Arguments

dat	A long-format triangulation dataset
types	A character vector of type values to flip (e.g. c("OR", "RR"))

Value

A modified dataset with adjusted y_i and d values

 tri_to_long

Convert data from wide to long format

Description

Reshapes domain-level bias columns from wide format (e.g., d_{1j} , d_{1t} , d_{1d}) to long format. Assumes columns follow naming pattern like d_{1j} , d_{1t} , d_{1d} , d_{2j} , etc.

Usage

```
tri_to_long(dat)
```

Arguments

dat	A data frame in wide format
-----	-----------------------------

Value

A long-format data frame with columns: domain, j, t, d

tri_to_wide	<i>Convert data from long to wide format</i>
-------------	--

Description

Reshapes domain-level bias columns from long format to wide format (e.g., dlj, dlt, dld). Assumes long format with columns 'domain', 'j', 't', 'd'.

Usage

```
tri_to_wide(dat)
```

Arguments

dat A long-format triangulate dataset

Value

A wide-format data frame

Index

* datasets

- beta_carotene_data, 2
- dat_bias, 3
- dat_bias_values, 3
- dat_ind, 3
- dat_ind_values, 4

beta_carotene_data, 2

dat_bias, 3

dat_bias_values, 3

dat_ind, 3

dat_ind_values, 4

interactive_bias_plot, 4

tri_absolute_direction, 5

tri_absolute_direction_invert, 5

tri_absolute_direction_quick, 6

tri_append_bias, 6

tri_append_indirect, 7

tri_calculate_adjusted_estimates, 7

tri_dat_check, 8

tri_plot_bias_direction, 8

tri_prep_data, 9

tri_swap_effect_direction, 10

tri_to_long, 10

tri_to_wide, 11