

Package: confMeta (via r-universe)

May 22, 2026

Title Confidence Curves and P-Value Functions for Meta-Analysis

Version 0.1.1

Description Provides tools for the combination of individual study results in meta-analyses using 'p-value' functions. Implements various combination methods including those by Fisher, Stouffer, Tippett, Edgington along with weighted generalizations. Contains functionality for the visualization and calculation of confidence curves and drapery plots to summarize evidence across studies.

License GPL (>= 3)

Language en-US

BugReports <https://github.com/SaveFonta/confMeta/issues>

URL <https://github.com/SaveFonta/confMeta>

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports ggplot2, meta, metafor, patchwork, ReplicationSuccess, scales, stats, rlang

Suggests knitr, rmarkdown, bayesmeta, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Config/pak/sysreqs cmake make libicu-dev libxml2-dev libx11-dev

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

Date/Publication 2026-05-22 13:04:00 UTC

RemoteUrl <https://github.com/savefonta/confmeta>

RemoteRef HEAD

RemoteSha ed8e139d00988029977dec6b6b924ca4b1a11448

Contents

autoplot.confMeta	2
confMeta	4
estimate_tau2	7
p_edgington	9
p_edgington_w	12
p_fisher	15
p_hmean	18
p_pearson	20
p_stouffer	22
p_tippett	24
p_value_functions	26
p_wilkinson	27

Index	30
--------------	-----------

autoplot.confMeta	<i>Visualizations of confMeta objects</i>
-------------------	---

Description

Plots one or more confMeta objects. This function can create two types of plots: the p -value function plot (also known as drapery plot) and the forest plot. This allows for direct visual comparison of different p -value functions.

Optionally, a Bayesian meta-analysis object created with the `bayesmeta::bayesmeta()` function can be supplied via the `bayesmeta` argument. When provided, its posterior summary is displayed as an additional diamond at the bottom of the forest plot for comparison with the frequentist methods

Important Note: If supplying a Bayesian meta-analysis object via the `bayesmeta` argument, this function explicitly extracts the 95% credible intervals. If the Bayesian model was fit using a different credible level, the function will crash.

Usage

```
## S3 method for class 'confMeta'
autoplot(
  ...,
  type = c("p", "forest"),
  diamond_height = 0.5,
  v_space = 1.5,
  scale_diamonds = TRUE,
  show_studies = TRUE,
  drapery = TRUE,
  reference_methods_p = c("re"),
  reference_methods_forest = c("re", "hk", "hc"),
  ref_labels = NULL,
  xlim = NULL,
```

```

xlim_p = NULL,
xlim_forest = NULL,
same_xlim = TRUE,
xlab = NULL,
n_breaks = 7,
bayesmeta = NULL,
n_points = 1000
)

```

Arguments

...	One or more objects of class confMeta.
type	A character vector of length 1 or 2. Indicates what type of plot should be returned. Accepted values are "p", "forest", or both. Defaults to c("p", "forest").
diamond_height	Numeric scalar. Indicates the maximal possible height of the diamonds in the forest plot. Defaults to 0.5. This argument is only relevant if type contains "forest" and will be ignored otherwise.
v_space	Numeric scalar. Indicates the vertical space between two diamonds in the forest plot. Defaults to 1.5. This argument is only relevant if type contains "forest" and will be ignored otherwise.
scale_diamonds	Logical. If TRUE (default), the diamond is rescaled to the interval [0, 1] in cases where the maximum of the p-value function is not equal to 1. This argument is only relevant if type contains "forest" and will be ignored otherwise.
show_studies	Logical. If TRUE (default), the forest plot shows the confidence intervals for the individual effect estimates. Otherwise, the intervals are suppressed. This argument is only relevant if type contains "forest" and will be ignored otherwise.
drapery	Logical. If TRUE (default), individual study effects are represented as drapery plots in the p-value function plot. If FALSE the studies are represented by a simple vertical line at their effect estimates. This argument is only relevant if type contains "p" and will be ignored otherwise.
reference_methods_p	Character vector controlling which reference meta-analysis methods are shown as baseline curves in the p-value function plot. Valid options are "fe", "re", or c("fe", "re") for both. Defaults to "re".
reference_methods_forest	Character vector of length 1 to 4. Specifies which reference meta-analysis methods should be shown as diamonds in the forest plot. Valid options are any subset of c("fe", "re", "hk", "hc"), which correspond to: <ul style="list-style-type: none"> • "fe": fixed-effect meta-analysis • "re": random-effects meta-analysis • "hk": Hartung-Knapp adjustment • "hc": Henmi-Copas adjustment Defaults to c("re", "hk", "hc").

ref_labels	A named character vector to customize the labels of the reference methods in the plot legends and axes (e.g., <code>c("fe" = "Fixed-effect (IV)", "re" = "Random-effects (DL)")</code>). Defaults to NULL.
xlim	Numeric vector of length 2. Global limits for the x-axis. If NULL, limits are calculated automatically.
xlim_p	Numeric vector of length 2. Specific x-axis limits for the <i>p</i> -value plot. Overrides <code>xlim</code> .
xlim_forest	Numeric vector of length 2. Specific x-axis limits for the forest plot. Overrides <code>xlim</code> .
same_xlim	Logical. If TRUE, forces the <i>p</i> -value plot to use the same x-axis limits as the forest plot. Defaults to TRUE.
xlab	Character string. Label for the x-axis. Defaults to NULL (renders as μ).
n_breaks	Numeric. Approximate number of tick marks to display on the x-axis. Defaults to 7.
bayesmeta	An object of class <code>bayesmeta</code> , typically created using <code>bayesmeta::bayesmeta()</code> . When provided, the posterior median (or mean) and 95% credible interval are displayed as an additional diamond at the bottom of the forest plot. Has no effect on the <i>p</i> -value plot. Defaults to NULL.
n_points	Numeric. Number of points used to create the <i>p</i> -value plot. Higher values (e.g., 10000) yield higher resolution but take longer to render. Defaults to 1000.

Value

An object of class `ggplot` containing the specified plot(s).

confMeta	<i>Creating confMeta objects</i>
----------	----------------------------------

Description

Function to create objects of class `confMeta`. This is the main class within the package. For an overview of available methods run `methods(class = "confMeta")`.

Usage

```
confMeta(
  estimates,
  SEs,
  study_names = NULL,
  conf_level = 0.95,
  fun,
  fun_name = NULL,
  w = NULL,
  MH = FALSE,
  table_2x2 = NULL,
```

```

measure = NULL,
method.tau.re = "REML",
method.tau.hk = "REML",
method.tau.het = "REML",
adhoc.hakn.ci = "IQWiG6",
...
)

```

Arguments

estimates	A vector containing the individual effect estimates. These should be on a scale where they are approximately normally distributed (e.g., log odds ratios, log risk ratios, mean differences). Must be of the same length as SEs and coercible to type double.
SEs	The standard errors of the individual effect estimates. Must be of the same length as estimates and coercible to type double.
study_names	Optional vector of study identifiers. If NULL (the default) names ("Study 1", "Study 2", ...) are used. Otherwise a vector that can be coerced to type character. Must be of the same length as arguments estimates and SEs.
conf_level	The confidence level (numeric scalar between 0 and 1).
fun	A function that combines estimates and SEs into a combined p -value. The function must accept arguments estimates, SEs, and mu. Any additional arguments that fun accepts can be passed via ... See p_value_functions for supported methods.
fun_name	A character vector of length 1. Label for fun used in <code>autoplot.confMeta</code> . Defaults to the name of the function passed to fun.
w	Numeric vector of weights for the studies. Must be of the same length as estimates and SEs, finite, and non-negative. If NULL (the default), equal weights are assumed. Used by weighted p -value combination functions.
MH	Logical. If TRUE, the fixed-effect comparison method will use Mantel-Haenszel pooling instead of the standard inverse-variance method. Requires table_2x2 and measure to be provided. Default is FALSE.
table_2x2	A data frame containing the 2x2 contingency table data for Mantel-Haenszel pooling. Required if MH = TRUE. Must contain columns: ai (events in experimental), bi (non-events in experimental), ci (events in control), di (non-events in control), n1i (total experimental), and n2i (total control).
measure	A character string indicating the effect measure to be used for Mantel-Haenszel pooling (e.g., "OR", "RR", "RD"). Required if MH = TRUE.
method.tau.re	Character string indicating which between-study variance estimator to use for random-effects meta-analysis (e.g., "PM", "REML", "DL"). Defaults to "REML". See <code>meta::metagen()</code> for all available choices.
method.tau.hk	Character string indicating which between-study variance estimator to use for the Hartung-Knapp meta-analysis. Defaults to "REML".
method.tau.het	Character string indicating which between-study variance estimator to use for calculating the general heterogeneity statistics in heterogeneity. Defaults to "REML".

`adhoc.hakn.ci` Character string indicating the variance correction method for the Hartung-Knapp confidence intervals (e.g., "IQWiG6", "HK"). Defaults to "IQWiG6". See `meta::metagen()` for choices.

... Additional arguments passed to `fun`.

Value

An S3 object of class `confMeta` containing the following elements:

- `estimates`, `SEs`, `w`, `study_names`, `conf_level`: Values used to build the object.
- `individual_cis`: Matrix of Wald-type confidence intervals for each study.
- `p_fun`: The p -value function used for combined inference.
- `joint_cis`: Combined confidence interval(s). Calculated by finding the values where the p -value function is larger than the significant level ($1 - \text{conf_level}$).
- `gamma`: The local minima within the range of the individual effect estimates. Column x refers to the mean μ and column y contains the corresponding p -value.
- `p_max`: The local maxima of the p -value function. Column x refers to the mean μ and column y contains the corresponding p -value.
- `p_0`: The value of the p -value at $\mu = 0$.
- `comparison_cis`: Combined confidence intervals calculated with other methods. These can be used for comparison purposes. Currently, these other methods are fixed effect (IV or Mantel-Haenszel), random effects (IV), Hartung & Knapp, and Henmi & Copas.
- `comparison_p_0`: The same as in element `p_0` but for the comparison methods (fixed effect, random effects, Hartung & Knapp, Henmi & Copas).
- `heterogeneity`: A data frame with columns `Q` (Cochran's Q), `p_Q` (p -value for Q), `I2` (I^2), `Tau` (τ), `I2_lower`, and `I2_upper`, computed using the estimator defined by `method.tau.het`.
- `table_2x2`: The 2×2 table data frame (if provided), otherwise `NULL`.

Confidence intervals

Individual confidence intervals are calculated as:

$$x_i \pm \Phi^{-1}(1 - \alpha/2) \cdot \sigma_i$$

where x_i are the estimates, σ_i the SEs, and $\alpha = 1 - \text{conf_level}$.

Combined confidence intervals are found by inverting the p -value function, identifying all μ where the p -value exceeds the significance level ($1 - \text{conf_level}$).

Also, the HK method uses `adhoc.hakn.ci = "IQWiG6"` by default, i.e., it uses variance correction if the HK confidence interval is narrower than the CI from the classic random effects model with the DerSimonian-Laird estimator (IQWiG, 2022).

See Also

[p_value_functions](#), [autoplot.confMeta](#)

Examples

```

# Simulate effect estimates and standard errors
set.seed(42)
n <- 5
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)
conf_level <- 0.95

# Construct a simple confMeta object using p_edgington as
# the p-value function
cm <- confMeta(
  estimates = estimates,
  SEs = SEs,
  conf_level = conf_level,
  fun = p_edgington,
  fun_name = "Edgington",
  input_p = "greater"
)

cm2 <- confMeta(
  estimates = estimates,
  SEs = SEs,
  conf_level = conf_level,
  fun = p_edgington_w,
  w = 1/SEs,
  fun_name = "Edgington (1/SE)",
  input_p = "greater"
)

# print the objects
cm
cm2

# Plot the objects
autoplot(cm, cm2, type = "p")           # p-value function plot
autoplot(cm, cm2, type = "forest")     # forest plot
autoplot(cm, cm2, type = c("p", "forest")) # both

```

estimate_tau2

Estimation of between-study heterogeneity

Description

estimate_tau2 estimates the between-study heterogeneity τ^2 using an additive model. The resulting parameter τ^2 is estimated through a call to `meta::metagen` with `TE = estimates` and `seTE = SEs`. Other arguments to `meta::metagen` can be passed via the `...` argument. If no arguments are passed via `...`, the following defaults are applied.

- `sm = "MD"`

- `method.tau = "REML"`
- `control = list(maxiter = 1e5, stepadj = 0.25)`

However, each of these defaults can be overwritten via

`estimate_phi` estimates the between-study heterogeneity ϕ using a multiplicative model. The function is a modified version of Page 3 of:

Accounting for heterogeneity in meta-analysis using a multiplicative model – an empirical study, Mawdsley D. et al. 2016. [doi:10.1002/jrsm.1216](https://doi.org/10.1002/jrsm.1216)

Usage

```
estimate_tau2(estimates, SEs, ...)
```

```
estimate_phi(estimates, SEs)
```

Arguments

<code>estimates</code>	Numeric vector of study-level effect estimates.
<code>SEs</code>	Numeric vector of corresponding standard errors.
<code>...</code>	Further arguments that are passed to <code>meta::metagen()</code> .

Value

For `estimate_tau2`: the estimated heterogeneity parameter τ^2 , a numeric vector of length 1.

For `estimate_phi`: the estimated heterogeneity parameter ϕ , a numeric vector of length 1.

Examples

```
# Example estimates and std. errors
estimates <- c(0.21, 0.53, 0.24, 0.32, 0.19)
SEs <- c(0.19, 0.39, 0.7, 1, 0.97)

# Estimating heterogeneity using the additive model
estimate_tau2(estimates = estimates, SEs = SEs)
# Example estimates and std. errors
estimates <- c(0.21, 0.53, 0.24, 0.32, 0.19)
SEs <- c(0.19, 0.39, 0.7, 1, 0.97)

estimate_phi(estimates = estimates, SEs = SEs)
```

p_edgington

*Edgington's method***Description**

Edgington's method for combining p -values across studies. The method forms a sum of individual study p -values and evaluates it against the exact or approximate null distribution.

Under the global null hypothesis, the null distribution of the sum is given by the Irwin–Hall distribution. For a weighted generalization of this procedure, see [p_edgington_w](#).

Usage

```
p_edgington(
  estimates,
  SEs,
  mu = 0,
  heterogeneity = "none",
  phi = NULL,
  tau2 = NULL,
  check_inputs = TRUE,
  input_p = "greater",
  output_p = "two.sided",
  approx = TRUE
)
```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
heterogeneity	One of <code>c("none", "additive", "multiplicative")</code> . If <code>heterogeneity = "none"</code> , p -values are returned for the passed SEs without any adaptation. If <code>heterogeneity = "additive"</code> , the standard errors are reassigned the value $\sqrt{SEs^2 + \text{tau2}}$ before computation of the p -values. If <code>heterogeneity = "multiplicative"</code> , the standard errors SEs are multiplied by $\sqrt{\text{phi}}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.
check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.

input_p	Type of study-level p -values used in the combination: "greater" (default), "less", or "two.sided". If "greater" or "less", one-sided p -values are combined.
output_p	Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. Note: To construct valid p -value functions, the <code>confMeta</code>
approx	Logical (default TRUE). If TRUE, use a normal approximation for the sum of p -values when $k \geq 12$ to avoid numerical overflow issues.

Details

The classical Edgington statistic is defined for k studies as

$$S = \sum_{i=1}^k p_i,$$

where p_i are individual study p -values. Under the global null hypothesis, each p_i is assumed to be uniformly distributed on $[0, 1]$.

Important note on orientation: Edgington's method is orientation-invariant. The combined p -value is symmetric with respect to the direction of the one-sided p -values (controlled by the `input_p` argument).

Specifically, computing the Edgington combined p -value for the "greater" alternative results in 1 minus the Edgington combined p -value for the "less" alternative.

Value

A numeric vector of combined p -values corresponding to each value of `mu`.

Null Distribution and Approximation

The combined p -value, p_E , is the probability of observing a sum less than or equal to S under the null hypothesis. This is computed in one of two ways:

- **Exact Method:** The function uses the exact Irwin-Hall distribution to compute the combined p -value:

$$p_E = \frac{1}{k!} \sum_{j=0}^{\lfloor S \rfloor} (-1)^j \binom{k}{j} (S - j)^k$$

- **Normal Approximation:** For a large number of studies ($k \geq 12$), the distribution of the sum is approximated by a Normal distribution with:

$$E[S] = \frac{k}{2}$$

$$\text{Var}(S) = \frac{k}{12}$$

Output p-value

The final output depends on the `output_p` and `input_p` arguments:

- If `output_p = "two.sided"` (the default) and the inputs are one-sided (`input_p` is "greater" or "less"), the function combines the one-sided p -values to obtain the intermediate combined p -value p_c , and returns a **symmetrized, two-sided p -value**: $p_{2s} = 2 \min(p_c, 1 - p_c)$.
- If `output_p = "one.sided"`, the function returns the inherently one-sided combined p -value p_c **directly, without symmetrization**.
- If `input_p` is "two.sided", the input p_i are already two-sided, and no further symmetrization is applied.

References

Edgington, E. S. (1972). An additive method for combining probability values from independent experiments. *The Journal of Psychology*, 80(2): 351-363. doi:10.1080/00223980.1972.9924813

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

Other p -value combination functions: [p_edgington_w\(\)](#), [p_fisher\(\)](#), [p_hmean\(\)](#), [p_pearson\(\)](#), [p_stouffer\(\)](#), [p_tippett\(\)](#), [p_wilkinson\(\)](#)

Examples

```
# Simulating estimates and standard errors
n <- 15
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)

# Set up a vector of means under the null hypothesis
mu <- seq(
  min(estimates) - 0.5 * max(SEs),
  max(estimates) + 0.5 * max(SEs),
  length.out = 100
)

# Using Edgington's method to calculate the combined p-value
p_edgington(
  estimates = estimates,
  SEs = SEs,
  mu = mu,
  heterogeneity = "none",
  output_p = "two.sided",
  input_p = "greater",
  approx = TRUE
)
```

p_edgington_w

*Weighted Edgington's method***Description**

Weighted generalization of Edgington's method for combining p -values across studies. The method forms a weighted sum of individual study p -values and evaluates it against the exact or approximate null distribution.

If all weights are equal, this reduces to the classical Edgington procedure, where the null distribution is given by the Irwin–Hall distribution.

Usage

```
p_edgington_w(
  estimates,
  SEs,
  mu = 0,
  heterogeneity = "none",
  phi = NULL,
  tau2 = NULL,
  check_inputs = TRUE,
  input_p = "greater",
  output_p = "two.sided",
  w = rep(1, length(estimates)),
  approx = TRUE,
  approx_rule = "neff",
  neff_cut = 12
)
```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \text{tau2}}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\text{phi}}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.

check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.
input_p	Type of study-level p -values used in the combination: "greater" (default), "less", or "two.sided". If "greater" or "less", one-sided p -values are combined.
output_p	Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. Note: To construct valid p -value functions, the confMeta
w	Numeric vector of nonnegative weights, same length as estimates. Defaults to equal weights.
approx	Logical (default TRUE). If TRUE, use a normal approximation for the weighted sum when the condition defined by approx_rule is met.
approx_rule	Rule for normal approximation: "n" uses the number of studies; "neff" (default) uses the effective sample size criterion (see Details).
neff_cut	Numeric threshold (default 12). If approx_rule="n", normal approximation is used when $n \geq 12$. If approx_rule="neff", normal approximation is used when $\ w\ _2^4 / \ w\ _4^4 \geq 12$.

Details

The weighted Edgington statistic is defined for k studies as

$$S = \sum_{i=1}^k w_i p_i,$$

where w_i are positive study weights and p_i are individual study p -values. Under the global null hypothesis, each p_i is assumed to be uniformly distributed on $[0, 1]$.

Value

A numeric vector of combined p -values corresponding to each value of mu.

Null Distribution and Approximation

The CDF of the test statistic S under the null, $F(t) = \Pr(S \leq t)$, is computed in one of two ways:

- **Exact Method:** The function uses the exact Barrow-Smith inclusion-exclusion formula to compute the CDF.

$$F(t) = \frac{1}{k! \prod_{i=1}^k w_i} \sum_{v \in \{0,1\}^k} (-1)^{\sum v_j} (t - w^T v)^k \mathbf{I}_{\{t - w^T v \geq 0\}}$$

This method is computationally intensive and is infeasible for $k > 18$ studies, at which point the function will stop with an error if approx = FALSE is used.

- **Normal Approximation:** For a large number of studies or sufficiently balanced weights, S is approximated by a Normal distribution with:

$$E[S] = \frac{1}{2} \sum_{i=1}^k w_i$$

$$\text{Var}(S) = \frac{1}{12} \sum_{i=1}^k w_i^2$$

Approximation Rule

The `approx = TRUE` argument enables the normal approximation, but it is only used if a condition (controlled by `approx_rule`) is met.

- `approx_rule = "n"`: Uses the approximation if the number of studies $n > \text{neff_cut}$.
- `approx_rule = "neff"` (default): Uses the approximation if the **effective sample size** $n_{\text{eff}} > \text{neff_cut}$.

The effective sample size is defined as:

$$n_{\text{eff}} = \frac{(\sum w_i^2)^2}{\sum w_i^4} = \frac{\|w\|_2^4}{\|w\|_4^4}$$

The default threshold `neff_cut = 12` is based on error bounds, which indicate the approximation is sufficiently accurate when this condition is met. This rule is more robust than `approx_rule = "n"` when weights are unbalanced.

Approximation Error

The `neff_cut` parameter directly controls the tolerance for the approximation error. **Edgeworth Approximation:** This provides an *approximation* of the error, which directly relates to the n_{eff} criterion used in this function:

$$\sup_{t \in \mathbb{R}} |F_n^w(t) - \Phi(t)| \approx \frac{\|\phi^{(3)}\|_{\infty}}{20} \frac{\|w\|_4^4}{\|w\|_2^4} \approx \frac{0.028}{n_{\text{eff}}}$$

The default `neff_cut = 12` is chosen based on this, as it corresponds to an approximate maximum error of $0.028/12 \approx 0.0023$. If you change `neff_cut`, you are changing this error tolerance.

Output p-value

The final output depends on the `output_p` and `input_p` arguments:

- If `output_p = "two.sided"` (the default) and the inputs are one-sided (`input_p` is "greater" or "less"), the function combines the one-sided p -values to obtain the intermediate combined p -value p_c , and returns a **symmetrized, two-sided p -value**: $p_{2s} = 2 \min(p_c, 1 - p_c)$.
- If `output_p = "one.sided"`, the function returns the inherently one-sided combined p -value p_c **directly, without symmetrization**.
- If `input_p` is "two.sided", the input p_i are already two-sided, and no further symmetrization is applied.

References

- Edgington, E. S. (1972). An additive method for combining probability values from independent experiments. *The Journal of Psychology*, 80(2): 351-363. doi:10.1080/00223980.1972.9924813
- Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26
- Barrow, D. L., & Smith, P. W. (1979). Spline notation applied to a volume problem. *The American Mathematical Monthly*, 86(1): 50-51. doi:10.1080/00029890.1979.11994730

See Also

Other p -value combination functions: [p_edgington\(\)](#), [p_fisher\(\)](#), [p_hmean\(\)](#), [p_pearson\(\)](#), [p_stouffer\(\)](#), [p_tippett\(\)](#), [p_wilkinson\(\)](#)

Examples

```
n <- 10
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)
weights <- 1/SEs

p_edgington_w(
  estimates = estimates,
  SEs = SEs,
  mu = 0,
  output_p = "two.sided",
  input_p = "greater",
  w = weights,
  approx = TRUE,
  approx_rule = "neff"
)
```

p_fisher

Fisher's method

Description

Fisher's method for combining p -values across studies. The method transforms the individual study p -values using the natural logarithm and evaluates the sum against a chi-squared distribution.

Usage

```
p_fisher(
  estimates,
  SEs,
  mu = 0,
  heterogeneity = "none",
  phi = NULL,
```

```

    tau2 = NULL,
    check_inputs = TRUE,
    input_p = "greater",
    output_p = "two.sided"
  )

```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \text{tau2}}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\text{phi}}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.
check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.
input_p	Type of study-level p -values used in the combination: "greater" (default), "less", or "two.sided". If "greater" or "less", one-sided p -values are combined.
output_p	Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. Note: To construct valid p -value functions, the confMeta

Details

The Fisher test statistic for k studies is defined as:

$$f = -2 \sum_{i=1}^k \log(p_i)$$

Under the global null hypothesis, each p_i is assumed to be uniformly distributed on $[0, 1]$. The test statistic f therefore follows a chi-squared distribution with $2k$ degrees of freedom: χ_{2k}^2 . The combined p -value, p_F , is calculated as the probability of observing a value strictly greater than f from this distribution:

$$p_F = \Pr(\chi_{2k}^2 > f)$$

Important note on orientation: Unlike Edgington's method, Fisher's method is *not* orientation-invariant. The combined p -value depends on the direction of the one-sided p -values (controlled by the input_p argument).

Specifically, Fisher's and Pearson's methods are mirrored. Computing the Fisher combined p -value for the "greater" alternative is equal to 1 minus the Pearson combined p -value for the "less" alternative.

Value

A numeric vector of combined p -values corresponding to each value of μ .

Output p-value

The final output depends on the `output_p` and `input_p` arguments:

- If `output_p = "two.sided"` (the default) and the inputs are one-sided (`input_p` is "greater" or "less"), the function combines the one-sided p -values to obtain the intermediate combined p -value p_c , and returns a **symmetrized, two-sided p -value**: $p_{2s} = 2 \min(p_c, 1 - p_c)$.
- If `output_p = "one.sided"`, the function returns the inherently one-sided combined p -value p_c **directly, without symmetrization**.
- If `input_p` is "two.sided", the input p_i are already two-sided, and no further symmetrization is applied.

References

Fisher R.A. *Statistical Methods for Research Workers*. 4th ed. Oliver & Boyd; 1932. doi:10.1093/oso/9780198522294.002.0003

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

Other p -value combination functions: [p_edgington\(\)](#), [p_edgington_w\(\)](#), [p_hmean\(\)](#), [p_pearson\(\)](#), [p_stouffer\(\)](#), [p_tippett\(\)](#), [p_wilkinson\(\)](#)

Examples

```
# Simulating estimates and standard errors
n <- 15
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)

# Set up a vector of means under the null hypothesis
mu <- seq(
  min(estimates) - 0.5 * max(SEs),
  max(estimates) + 0.5 * max(SEs),
  length.out = 100
)

# Using Fisher's method to calculate the combined p-value
p_fisher(
  estimates = estimates,
  SEs = SEs,
```

```

    mu = mu,
    heterogeneity = "none",
    output_p = "two.sided",
    input_p = "greater"
  )

```

p_hmean

Harmonic mean method

Description

Combines study-level results using the harmonic mean combination method

Usage

```

p_hmean(
  estimates,
  SEs,
  mu = 0,
  phi = NULL,
  tau2 = NULL,
  heterogeneity = "none",
  check_inputs = TRUE,
  w = rep(1, length(estimates)),
  distr = "chisq"
)

```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.
heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \text{tau2}}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\text{phi}}$ before computation of the p -values. Defaults to "none".
check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.

w	Numeric vector of nonnegative weights, same length as estimates. Defaults to equal weights.
distr	The distribution to use for the calculation of the p-value. Currently, the options are "f" (F-distribution) and "chisq" (Chi-squared distribution). Defaults to "chisq".

Details

The harmonic mean statistic for k studies is defined as

$$x^2 = \frac{\sum_{i=1}^k \sqrt{w_i}}{\sum_{i=1}^k w_i / z_i^2},$$

where z_i and w_i are individual study z -values and weights, respectively. Under the global null hypothesis, each z_i is assumed to follow a standard normal distribution. The harmonic mean statistic then follows a chi-squared distribution with one degree of freedom. The combined p -value is calculated as the probability of observing a value equal or greater than x^2 from this distribution:

$$p_H = \Pr(\chi_1^2 > x^2)$$

Value

A numeric vector of combined p -values corresponding to each value of μ .

References

Held, L. (2020). The harmonic mean chi-squared test to substantiate scientific findings. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 69:697-708. doi:10.1111/rssc.12410

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

Other p -value combination functions: [p_edgington\(\)](#), [p_edgington_w\(\)](#), [p_fisher\(\)](#), [p_pearson\(\)](#), [p_stouffer\(\)](#), [p_tippett\(\)](#), [p_wilkinson\(\)](#)

Examples

```
estimates <- c(0.5, 0.8, 0.3)
SEs <- c(0.1, 0.2, 0.1)
p_hmean(estimates, SEs, mu = 0, distr = "f")
```

p_pearson

*Pearson's method***Description**

Pearson's method for combining p -values across studies. The method transforms the complement of the individual study p -values using the natural logarithm and evaluates the sum against a chi-squared distribution.

Usage

```
p_pearson(
  estimates,
  SEs,
  mu = 0,
  heterogeneity = "none",
  phi = NULL,
  tau2 = NULL,
  check_inputs = TRUE,
  input_p = "greater",
  output_p = "two.sided"
)
```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \tau2}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\phi}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.
check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.
input_p	Type of study-level p -values used in the combination: "greater" (default), "less", or "two.sided". If "greater" or "less", one-sided p -values are combined.

output_p Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. **Note:** To construct valid p -value functions, the `confMeta`

Details

The Pearson test statistic for k studies is defined as:

$$g = -2 \sum_{i=1}^k \log(1 - p_i)$$

Under the global null hypothesis, each p_i is assumed to be uniformly distributed on $[0, 1]$. The test statistic g therefore follows a chi-squared distribution with $2k$ degrees of freedom: χ_{2k}^2 . The combined p -value, p_P , is calculated as the probability of observing a value less than or equal to g from this distribution:

$$p_P = \Pr(\chi_{2k}^2 \leq g)$$

Important note on orientation: Unlike Edgington's method, Pearson's method is *not* orientation-invariant. The combined p -value depends on the direction of the one-sided p -values (controlled by the `input_p` argument).

Specifically, Pearson's and Fisher's methods are mirrored. Computing the Pearson combined p -value for the "greater" alternative is equal to 1 minus the Fisher combined p -value for the "less" alternative.

Value

A numeric vector of combined p -values corresponding to each value of μ .

Output p-value

The final output depends on the `output_p` and `input_p` arguments:

- If `output_p` = "two.sided" (the default) and the inputs are one-sided (`input_p` is "greater" or "less"), the function combines the one-sided p -values to obtain the intermediate combined p -value p_c , and returns a **symmetrized, two-sided p -value**: $p_{2s} = 2 \min(p_c, 1 - p_c)$.
- If `output_p` = "one.sided", the function returns the inherently one-sided combined p -value p_c **directly, without symmetrization**.
- If `input_p` is "two.sided", the input p_i are already two-sided, and no further symmetrization is applied.

References

Pearson K. On a method of determining whether a sample of size n supposed to have been drawn from a parent population having a known probability integral has probably been drawn at random. *Biometrika*, 25:379-410, 1933. doi:10.2307/2332290

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

Other p-value combination functions: [p_edgington\(\)](#), [p_edgington_w\(\)](#), [p_fisher\(\)](#), [p_hmean\(\)](#), [p_stouffer\(\)](#), [p_tippett\(\)](#), [p_wilkinson\(\)](#)

Examples

```
# Simulating estimates and standard errors
n <- 15
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)

# Set up a vector of means under the null hypothesis
mu <- seq(
  min(estimates) - 0.5 * max(SEs),
  max(estimates) + 0.5 * max(SEs),
  length.out = 100
)

# Using Pearson's method to calculate the combined p-value
p_pearson(
  estimates = estimates,
  SEs = SEs,
  mu = mu,
  heterogeneity = "none",
  output_p = "two.sided",
  input_p = "greater"
)
```

p_stouffer

Stouffer's method

Description

Stouffer's method for combining p -values across studies

Usage

```
p_stouffer(
  estimates,
  SEs,
  mu = 0,
  heterogeneity = "none",
  phi = NULL,
  tau2 = NULL,
  check_inputs = TRUE,
  w = 1/SEs,
  output_p = "two.sided"
)
```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \text{tau2}}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\text{phi}}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.
check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.
w	Numeric vector of study weights. Defaults to 1/SEs producing the same p -value as meta-analysis
output_p	Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. Note: To construct valid p -value functions, the confMeta

Details

Stouffer's z -statistic for k studies is defined as

$$z = \frac{\sum_{i=1}^k w_i z_i}{\sqrt{\sum_{i=1}^k w_i^2}},$$

where z_i and w_i are individual study z -values and weights, respectively. Under the global null hypothesis, each z_i is assumed to follow a standard normal distribution. Stouffer's z -statistic then also follows a standard normal distribution. The combined p -value is calculated as the probability of observing a value equal or greater than z from this distribution:

$$p_S = \Pr(Z > z)$$

Value

A numeric vector of combined p -values corresponding to each value of mu.

References

- Stouffer SA, et al. *The American Soldier*. Princeton University Press, 1949. [doi:10.2307/2572105](https://doi.org/10.2307/2572105)
- Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. [doi:10.1017/rsm.2025.26](https://doi.org/10.1017/rsm.2025.26)

See Also

Other p-value combination functions: [p_edgington\(\)](#), [p_edgington_w\(\)](#), [p_fisher\(\)](#), [p_hmean\(\)](#), [p_pearson\(\)](#), [p_tippett\(\)](#), [p_wilkinson\(\)](#)

Examples

```
estimates <- c(0.1, 0.2, 0.3)
SEs <- c(0.05, 0.05, 0.1)
p_stouffer(estimates, SEs, mu = 0, heterogeneity = "none")
```

p_tippett

Tippett's method

Description

Tippett's method for combining p -values across studies. This method evaluates the minimum individual study p -value.

Usage

```
p_tippett(
  estimates,
  SEs,
  mu = 0,
  heterogeneity = "none",
  phi = NULL,
  tau2 = NULL,
  check_inputs = TRUE,
  input_p = "greater",
  output_p = "two.sided"
)
```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).
heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \tau^2}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\phi}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.

check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.
input_p	Type of study-level p -values used in the combination: "greater" (default), "less", or "two.sided". If "greater" or "less", one-sided p -values are combined.
output_p	Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. Note: To construct valid p -value functions, the confMeta

Details

The Tippett combined p -value, p_T , for k studies is defined directly as:

$$p_T = 1 - (1 - \min\{p_1, \dots, p_k\})^k$$

Under the global null hypothesis, each p_i is assumed to be uniformly distributed on $[0, 1]$.

Important note on orientation: Unlike Edgington's method, Tippett's method is *not* orientation-invariant. The combined p -value depends on the direction of the one-sided p -values (controlled by the input_p argument).

Specifically, Tippett's and Wilkinson's methods are mirrored. Computing the Tippett combined p -value for the "greater" alternative is equal to 1 minus the Wilkinson combined p -value for the "less" alternative.

Value

A numeric vector of combined p -values corresponding to each value of mu.

Output p-value

The final output depends on the output_p and input_p arguments:

- If output_p = "two.sided" (the default) and the inputs are one-sided (input_p is "greater" or "less"), the function combines the one-sided p -values to obtain the intermediate combined p -value p_c , and returns a **symmetrized, two-sided p -value**: $p_{2s} = 2 \min(p_c, 1 - p_c)$.
- If output_p = "one.sided", the function returns the inherently one-sided combined p -value p_c **directly, without symmetrization**.
- If input_p is "two.sided", the input p_i are already two-sided, and no further symmetrization is applied.

References

Tippett LHC. *Methods of Statistics*. Williams Norgate; 1931. doi:10.2307/3606890

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

Other p-value combination functions: [p_edgington\(\)](#), [p_edgington_w\(\)](#), [p_fisher\(\)](#), [p_hmean\(\)](#), [p_pearson\(\)](#), [p_stouffer\(\)](#), [p_wilkinson\(\)](#)

Examples

```
# Simulating estimates and standard errors
n <- 15
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)

# Set up a vector of means under the null hypothesis
mu <- seq(
  min(estimates) - 0.5 * max(SEs),
  max(estimates) + 0.5 * max(SEs),
  length.out = 100
)

# Using Tippett's method to calculate the combined p-value
p_tippett(
  estimates = estimates,
  SEs = SEs,
  mu = mu,
  heterogeneity = "none",
  output_p = "two.sided",
  input_p = "greater"
)
```

p_value_functions

Overview of p-value combination functions

Description

The `p_value_functions` family is useful to combine individual study effect estimates and standard errors into a single meta-analytical p -value function. These functions are the building blocks for the [confMeta](#) function.

Details

Vectorization over mu: All of the p -value functions in this package are vectorized over the `mu` argument. This means that you can pass a sequence of null values (e.g., `seq(-2, 2, length.out = 1000)`) and the function will evaluate the combined p -value for every point. This is specifically designed to construct p -value functions and invert the tests to find confidence intervals at any desired level.

Available Methods:

- [p_edgington](#): Classical Edgington method (Sum of p -values).
- [p_edgington_w](#): Weighted Edgington method.

- [p_fisher](#): Fisher's method (Log-product of p -values).
- [p_pearson](#): Pearson's method (Log-product of complements).
- [p_tippett](#): Tippett's method (Minimum p -value).
- [p_wilkinson](#): Wilkinson's method (Maximum p -value).
- [p_stouffer](#): Stouffer's method.
- [p_hmean](#): Harmonic mean method.

References

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

[confMeta](#)

p_wilkinson

Wilkinson's method

Description

Wilkinson's method for combining p -values across studies. This method evaluates the maximum individual study p -value.

Usage

```
p_wilkinson(  
  estimates,  
  SEs,  
  mu = 0,  
  heterogeneity = "none",  
  phi = NULL,  
  tau2 = NULL,  
  check_inputs = TRUE,  
  input_p = "greater",  
  output_p = "two.sided"  
)
```

Arguments

estimates	Numeric vector of study-level effect estimates.
SEs	Numeric vector of corresponding standard errors.
mu	Numeric scalar or vector of null values for the overall effect (default: 0).

heterogeneity	One of c("none", "additive", "multiplicative"). If heterogeneity = "none", p -values are returned for the passed SEs without any adaptation. If heterogeneity = "additive", the standard errors are reassigned the value $\sqrt{SEs^2 + \text{tau2}}$ before computation of the p -values. If heterogeneity = "multiplicative", the standard errors SEs are multiplied by $\sqrt{\text{phi}}$ before computation of the p -values. Defaults to "none".
phi	A numeric vector of length 1. Must be finite and larger than 0. The square root of the argument is used to scale the standard errors.
tau2	A numeric vector of length 1. Additive heterogeneity parameter.
check_inputs	Either TRUE (default) or FALSE. Indicates whether or not to check the input arguments. The idea of this argument is that if the function is called a large amount of times in an automated manner as for example in simulations, performance might be increased by not checking inputs in every single iteration. However, setting the argument to FALSE might be dangerous.
input_p	Type of study-level p -values used in the combination: "greater" (default), "less", or "two.sided". If "greater" or "less", one-sided p -values are combined.
output_p	Character string specifying the combined p -value type: "two.sided" (default) or "one.sided". This controls whether the final combined p -value is symmetrized. Note: To construct valid p -value functions, the <code>confMeta</code>

Details

The Wilkinson combined p -value for k studies is defined as:

$$p_W = \max\{p_1, \dots, p_k\}^k$$

Under the global null hypothesis, each p_i is assumed to be uniformly distributed on $[0, 1]$.

Important note on orientation: Unlike Edgington's method, Wilkinson's method is *not* orientation-invariant. The combined p -value depends on the direction of the one-sided p -values (controlled by the `input_p` argument).

Specifically, Wilkinson's and Tippett's methods are mirrored. Computing the Wilkinson combined p -value for the "greater" alternative is equal to 1 minus the Tippett combined p -value for the "less" alternative.

Value

A numeric vector of combined p -values corresponding to each value of μ .

Output p-value

The final output depends on the `output_p` and `input_p` arguments:

- If `output_p` = "two.sided" (the default) and the inputs are one-sided (`input_p` is "greater" or "less"), the function combines the one-sided p -values to obtain the intermediate combined p -value p_c , and returns a **symmetrized, two-sided p -value**: $p_{2s} = 2 \min(p_c, 1 - p_c)$.
- If `output_p` = "one.sided", the function returns the inherently one-sided combined p -value p_c **directly, without symmetrization**.

- If input_p is "two.sided", the input p_i are already two-sided, and no further symmetrization is applied.

References

Wilkinson B. A statistical consideration in psychological research. *Psychological Bulletin*, 48(2):156-158, 1951. doi:10.1037/h0059111

Held, L, Hofmann, F, Pawel, S. (2025). A comparison of combined p -value functions for meta-analysis. *Research Synthesis Methods*, 16:758-785. doi:10.1017/rsm.2025.26

See Also

Other p -value combination functions: [p_edgington\(\)](#), [p_edgington_w\(\)](#), [p_fisher\(\)](#), [p_hmean\(\)](#), [p_pearson\(\)](#), [p_stouffer\(\)](#), [p_tippett\(\)](#)

Examples

```
# Simulating estimates and standard errors
n <- 15
estimates <- rnorm(n)
SEs <- rgamma(n, 5, 5)

# Set up a vector of means under the null hypothesis
mu <- seq(
  min(estimates) - 0.5 * max(SEs),
  max(estimates) + 0.5 * max(SEs),
  length.out = 100
)

# Using Wilkinson's method to calculate the combined p-value
p_wilkinson(
  estimates = estimates,
  SEs = SEs,
  mu = mu,
  heterogeneity = "none",
  output_p = "two.sided",
  input_p = "greater"
)
```

Index

* p-value combination functions

p_edgington, 9

p_edgington_w, 12

p_fisher, 15

p_hmean, 18

p_pearson, 20

p_stouffer, 22

p_tippett, 24

p_wilkinson, 27

autoplot.confMeta, 2, 5, 6

confMeta, 4, 10, 13, 16, 21, 23, 25–28

estimate_phi (estimate_tau2), 7

estimate_tau2, 7

p_edgington, 9, 15, 17, 19, 22, 24, 26, 29

p_edgington_w, 9, 11, 12, 17, 19, 22, 24, 26,
29

p_fisher, 11, 15, 15, 19, 22, 24, 26, 27, 29

p_hmean, 11, 15, 17, 18, 22, 24, 26, 27, 29

p_pearson, 11, 15, 17, 19, 20, 24, 26, 27, 29

p_stouffer, 11, 15, 17, 19, 22, 22, 26, 27, 29

p_tippett, 11, 15, 17, 19, 22, 24, 24, 27, 29

p_value_functions, 5, 6, 26

p_wilkinson, 11, 15, 17, 19, 22, 24, 26, 27, 27