

Package ‘ORCI.Welch’

August 27, 2026

Type Package

Title Approximate Odds Ratio Confidence Intervals with Welch's Adjustments

Version 0.1.1

Author Cosette Clinton [aut],
McKenna Morgan [ctb],
Kimihiro Noguchi [aut, cre]

Maintainer Kimihiro Noguchi <kimihiro.noguchi@wwu.edu>

Description Calculates approximate odds ratio confidence intervals with Welch's adjustments for a given dataset and ranks different odds ratio confidence intervals in terms of multiple metrics based on the dataset given. References for Welch's adjustments can be found at Pan (2002) <[doi:10.1016/S0167-9473\(01\)00107-4](https://doi.org/10.1016/S0167-9473(01)00107-4)> and Clinton, Morgan, and Noguchi (2026) <<https://meetings.ams.org/math/jmm2026/meetingapp.cgi/Paper/58471>>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 3.5)

Imports graphics

NeedsCompilation no

Config/roxygen2/version 8.1.0

Repository CRAN

Date/Publication 2026-08-27 10:00:12 UTC

Contents

barplot.ORCompareCounts	2
Doll_Hill_1950	4
Dube_2009	5
ORCompare_Counts	6
ORCompare_Sort	8

OR_Compare	9
OR_Welch	12
plot.ORCompareCounts	14
print.ORCompare	16
print.ORCompareCounts	17
print.ORCompareSort	18
Rake_2009	19
Sando_2008	21
summary.ORCompare	22
summary.ORCompareCounts	23
summary.ORCompareSort	24
Wei_2015	26
Index	28

barplot.ORCompareCounts	<i>Bar Graph Summary of Approximate Odds Ratio Confidence Interval Rankings</i>
-------------------------	---

Description

Plots objects of the class "ORCompareCounts" in stacked bar graphs to summarize the the top-performing approximate odds ratio confidence interval methods.

Usage

```
## S3 method for class 'ORCompareCounts'
barplot(height, ...)
```

Arguments

- height An object of class "ORCompareCounts".
- ... Additional arguments.

Details

This function generates stacked bar graphs showing how often each of the approximate odds ratio confidence interval methods implemented in this package appears within ranking groups.

Value

No return value, called for plotting objects of the class "ORCompareCounts".

See Also

[OR_Compare](#), [summary.ORCompare](#), [print.ORCompare](#).

Examples

```

#Testing with the data from Dube et al. (2009)
test_Dube <- OR_Compare(
  x = as.numeric(Dube_2009[1, 2:3]),
  n = as.numeric(Dube_2009[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
test_Dube_counts <- ORCompare_Counts(test_Dube)

oldpar <- par(mfrow = c(1,2))
par(mar = c(15, 4.1, 4.1, 2.1))
colors <- c("blue", "orange")
barplot(test_Dube_counts, xlab = "", ylab = "Count",
        main = "Stacked Bar Graph of the Ranking Groups",
        col = colors, las = 2, xlim = c(0, 6))
legend("topright",
      legend = test_Dube_counts$Rank_Group[c(1, 5)],
      col = colors, title = "Ranking Groups",
      lwd = 2, cex = 0.8)
par(oldpar)

#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
sim_results_counts <- ORCompare_Counts(sim_results)

oldpar <- par(mfrow = c(1,2))
par(mar = c(15, 4.1, 4.1, 2.1))
colors <- c("blue", "orange")
barplot(sim_results_counts, xlab = "", ylab = "Count",
        main = "Stacked Bar Graph of the Ranking Groups",
        col = colors, las=2, xlim=c(0,6))
legend("topright",
      legend = sim_results_counts$Rank_Group[c(1,5)],
      col = colors, title = "Ranking Groups",
      lwd = 2, cex = 0.8)
par(oldpar)

```

Doll_Hill_1950

*Smoking Status and Lung Cancer Case-Control Data***Description**

A 2-by-2 contingency table from Doll and Hill (1950) comparing lung cancer status between male smokers who smoke 25-49 cigarettes per day and male non-smokers.

Usage

Doll_Hill_1950

Format

A data frame with 2 rows and 3 variables:

Lung_Cancer_Status Lung cancer status ("Lung Cancer", "Control").

Heavy_Smoker Number of individuals smoking 25–49 cigarettes per day.

Non_Smoker Number of non-smokers.

Total Row Totals.

The contingency table is:

Lung Cancer Status	25–49 Cigarettes/Day	Non-Smoker	Total
Lung Cancer	136	2	138
Control	71	27	98
Total	207	29	236

Details

The data contains counts from a case-control study.

The estimated odds ratio (OR) is

$$OR \approx 25.86.$$

That is, males who smoke 25-49 cigarettes per day have 25.8592 times the odds of being in the lung cancer group compared to the control group, meaning that there is a strong association between smoking one or more packs of cigarettes a day (a pack of cigarettes generally has 25+ cigarettes) and lung cancer.

The proportion of lung cancer cases among heavy smokers is approximately 0.9855.

Source

Doll, C. and Hill, A. B. (1950). See [doi:10.1136/bmj.2.4682.739](https://doi.org/10.1136/bmj.2.4682.739).

References

Doll, R. and Hill, A. B. (1950). Smoking and carcinoma of the lung. *British Medical Journal*, 2, 739–748.

Examples

```
data(Doll_Hill_1950)
Doll_Hill_1950
n11 <- as.numeric(Doll_Hill_1950[1,2])
n12 <- as.numeric(Doll_Hill_1950[1,3])
n21 <- as.numeric(Doll_Hill_1950[2,2])
n22 <- as.numeric(Doll_Hill_1950[2,3])
OR <- (n11 * n22) / (n12 * n21)
OR
```

Dube_2009

Adverse Childhood Experiences and Lifetime Suicide Attempts

Description

A 2-by-2 contingency table from Dube et al. (2009) relating adverse childhood experiences (ACE) score to lifetime suicide attempts.

Usage

Dube_2009

Format

A data frame with 2 observations and 3 variables:

ACE_Score ACE category (" ≥ 7 ", "None").

Suicide_Attempt_Yes Number reporting a lifetime suicide attempt.

Suicide_Attempt_No Number reporting no lifetime suicide attempt.

Total Row Totals

Details

The data compares individuals with an ACE score of seven or more to individuals reporting no adverse childhood experiences.

The estimated odds ratio (OR) is

$$OR \approx 48.50.$$

That is, individuals with an Adverse Childhood Experiences (ACE) score of 7 or more have approximately 48.50 times the odds of reporting a lifetime suicide attempt compared to individuals with no reported adverse childhood experiences. This suggests a very strong positive association between high ACE scores and lifetime suicide attempts.

The proportion of suicide attempts among individuals with an ACE score of seven or more is approximately 0.3522.

Source

Dube, S. R., Anda, R. F., Felitti, V. J., Chapman, D. P., Williamson, D. F., and Giles, W. H. (2009). See [doi:10.1001/jama.286.24.3089](https://doi.org/10.1001/jama.286.24.3089).

References

Dube, S. R., Anda, R. F., Felitti, V. J., Chapman, D. P., Williamson, D. F., and Giles, W. H. (2009). Childhood abuse, household dysfunction, and the risk of attempted suicide throughout the life span. *JAMA*, 286(24), 3089–3096.

Examples

```
data(Dube_2009)
Dube_2009
n11 <- as.numeric(Dube_2009[1,2])
n12 <- as.numeric(Dube_2009[1,3])
n21 <- as.numeric(Dube_2009[2,2])
n22 <- as.numeric(Dube_2009[2,3])
OR <- (n11 * n22) / (n12 * n21)
OR
```

ORCompare_Counts	<i>Computing Descriptive Statistics of Approximate Odds Ratio Confidence Interval Rankings</i>
------------------	--

Description

Computes descriptive statistics of objects of the class "ORCompare" by creating a data frame of top-performing approximate odds ratio confidence interval methods.

Usage

```
ORCompare_Counts(object, rank.by = NULL, n = 5, m = 2, l = 4)
```

Arguments

object	An object of class "ORCompare".
rank.by	A character string or a vector of character strings specifying which metric(s) to rank by. The default option is NULL, in which case, every option below is used. The available options are: • "Coverage_Prob_All" • "Coverage_Prob_NoNA" • "Mean_Log_Length" • "Median_Log_Length" • "Mean_Orig_Length"

- "Median_Orig_Length"
 - "NA_Counts"
- n** An integer specifying the number of rankings in each group. The default option is 5.
- m** An integer specifying the number of ranking groups. The default option is 2.
- l** An integer specifying the number of methods to display. The default option is 4.

Details

This function calculates descriptive statistics of of the class "ORCompare" by giving counts (frequencies) of each approximate odds ratio confidence interval method appearing in the rankings.

Value

An "ORCompareCounts" class object.

See Also

[OR_Compare](#), [summary.ORCompareCounts](#), [print.ORCompareCounts](#), [plot.ORCompareCounts](#), [barplot.ORCompareCounts](#)

Examples

```
#Testing with the data from Wei et al. (2015)
test_Wei <- OR_Compare(
  x = as.numeric(Wei_2015[1, 2:3]),
  n = as.numeric(Wei_2015[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
ORCompare_Counts(test_Wei)
```

```
#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
ORCompare_Counts(sim_results, rank.by="Coverage_Prob_All",
  n = 3, m = 3, l = 5)
ORCompare_Counts(sim_results)
```

ORCompare_Sort	<i>Sorting a Summary of Approximate Odds Ratio Confidence Interval Rankings</i>
----------------	---

Description

Sorts the top-performing approximate odds ratio confidence interval methods for objects of the class "ORCompare".

Usage

```
ORCompare_Sort(x, top.n = 10, rank.by = NULL)
```

Arguments

x	An object of class "ORCompare" from OR_Compare .
top.n	An integer specifying how many of the top methods to extract. The default option is 10.
rank.by	A character string or a vector of character strings specifying which metric(s) to rank by. The default option is NULL, in which case, every option below is used. The available options are: • "Coverage_Prob_All" • "Coverage_Prob_NoNA" • "Mean_Log_Length" • "Median_Log_Length" • "Mean_Orig_Length" • "Median_Orig_Length" • "NA_Counts"

Details

This function sorts the best-performing approximate odds ratio confidence intervals according to the chosen metric(s). By default, these methods are ranked by how close the simulated coverage probability is to the nominal confidence level.

Value

An "ORCompareSort" class object.

See Also

[OR_Compare](#), [summary.ORCompareSort](#), [print.ORCompareSort](#).

Examples

```
#Testing with the data from Sando et al. (2008)
test_Sando <- OR_Compare(
  x = as.numeric(Sando_2008[1, 2:3]),
  n = as.numeric(Sando_2008[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
ORCompare_Sort(test_Sando)

#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
ORCompare_Sort(sim_results, top.n = 5, rank.by = "Median_Log_Length")
ORCompare_Sort(sim_results)
```

OR_Compare

*Comparing Different Approximate Odds Ratio Confidence Intervals
with Simulation*

Description

Evaluates and compares multiple approximate odds ratio confidence intervals using Monte Carlo simulation.

Usage

```
OR_Compare(
  x,
  n,
  alternative = c("two.sided", "less", "greater"),
  data.gen = c("Binomial", "Multinomial", "Poisson"),
  conf.level = 0.95,
  nsim = 10000,
```

```

    fix.seed = TRUE,
    seed.num = 1234
  )

```

Arguments

<code>x</code>	A numeric vector of length 2 giving the numbers of successes for the first and second row of the 2-by-2 contingency table.
<code>n</code>	A numeric vector of length 2 giving the total cell counts for the first and second row of the 2-by-2 contingency table.
<code>alternative</code>	A character string specifying the alternative hypothesis. It must be one of "two.sided" (default), "greater" or "less".
<code>data.gen</code>	A character string specifying the distribution to be used for data generation. The default option is "Binomial", which indicates that there are two independent binomial samples. If "Multinomial" is chosen, that indicates that the four cell counts follow a multinomial distribution. If "Poisson" is chosen, that indicates that there are four independent Poisson samples.
<code>conf.level</code>	A numeric specifying the confidence level. It must be a single number between 0 and 1. The default option is 0.95 for the 95% confidence interval.
<code>nsim</code>	A numeric specifying the number of datasets to be generated for the Monte Carlo simulation. The default option is 10000.
<code>fix.seed</code>	A Boolean specifying whether to fix seed for replication. The default option is TRUE, in which case, the seed is fixed using <code>seed.num</code> .
<code>seed.num</code>	A numeric specifying the seed number to be used for setting the seed when <code>fix.seed</code> is set to TRUE. The default option is 1234.

Details

This function evaluates and compares various approximate odds ratio confidence intervals in terms of the coverage probability, mean length, median length, mean log length, median log length, and the number of cases where exceptions (NAs) are produced. These approximate odds ratio confidence intervals vary in terms of the degrees of freedom adjustments, cell count adjustments, standard error calculations, and endpoint adjustment.

Value

An object of class "ORCompare" with the following columns:

DF The distribution assumed for the degrees of freedom calculation. See `data.gen` for details.

Cell_Count_Adj The cell count adjustment method used. See `adjust` in [OR_Welch](#) for details.

Std_Err The standard error transformation used. See `se.option` in [OR_Welch](#) for details.

Endpoint_Adj The endpoint adjustment method used. See `endpoint` in [OR_Welch](#) for details.

Coverage_Prob_All The coverage probability including the exceptional cases with an indefinite form for the odds ratio estimate.

Rank_Coverage_Prob_All Ranking of the odds ratio confidence intervals based on `Coverage_Prob_All`.

Coverage_Prob_NoNA Coverage probability excluding the exceptional cases with an indefinite form for the odds ratio estimate.

Rank_Coverage_Prob_NoNA Ranking of the odds ratio confidence intervals based on Coverage_Prob_NoNA.

Mean_Log_Length Mean log length of the odds ratio confidence intervals. It is applicable only if "two.sided" is chosen for the alternative hypothesis.

Rank_Mean_Log_Length Ranking of the odds ratio confidence intervals based on Mean_Log_Length.

Median_Log_Length Median log length of the odds ratio confidence intervals. It is applicable only if "two.sided" is chosen for the alternative hypothesis.

Rank_Median_Log_Length Ranking of the odds ratio confidence intervals based on Median_Log_Length.

Mean_Orig_Length Mean length of the odds ratio confidence intervals.

Rank_Mean_Orig_Length Ranking of the odds ratio confidence intervals based on Mean_Orig_Length.

Median_Orig_Length Median length of the odds ratio confidence intervals.

Rank_Median_Orig_Length Ranking of the odds ratio confidence intervals based on Median_Orig_Length.

NA_Counts Number of NA counts in the Monte Carlo simulation.

Rank_NA_Counts Ranking of the odds ratio confidence intervals based on NA_Counts.

See Also

See [OR_Welch](#) for computing approximate odds ratio confidence intervals with Welch's adjustments. Also see [ORCompare_Counts](#) and [ORCompare_Sort](#) for summarizing the results.

Examples

```
#Testing with the data from Doll and Hill (1950)
test_Doll_Hill <- OR_Compare(
  x = as.numeric(Doll_Hill_1950[1, 2:3]),
  n = as.numeric(Doll_Hill_1950[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
```

```
#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
```

OR_Welch

*Odds Ratio Confidence Interval with Welch Adjustment Options***Description**

Computes the odds ratio confidence interval, test statistic, and p-value based on Student's t distribution with Welch's degrees of freedom.

Usage

```
OR_Welch(
  x,
  n,
  alternative = c("two.sided", "less", "greater"),
  OR.null = 1,
  dist = c("Binomial", "Multinomial", "Poisson", "Asymptotic"),
  endpoint = FALSE,
  adjust = c("Agresti", "Gart", "Fagerland", "Woolf", "Custom"),
  adjust.custom = rep(0, 4),
  se.option = c("original", "sinh"),
  conf.level = 0.95
)
```

Arguments

x	A numeric vector of length 2 giving the numbers of successes for the first and second row of the 2-by-2 contingency table.
n	A numeric vector of length 2 giving the total cell counts for the first and second row of the 2-by-2 contingency table.
alternative	A character string specifying the alternative hypothesis. It must be one of "two.sided" (default), "greater" or "less".
OR.null	A numeric specifying the odds ratio value under the null hypothesis. The default value is 1.
dist	A character string specifying the distribution to be used: "Binomial" for t with Welch's adjustment assuming two independent binomial distributions, "Multinomial" for t with Welch's adjustment assuming a multinomial distribution, "Poisson" for t with Welch's adjustment assuming four independent Poisson distributions, and "Asymptotic" for no Welch's adjustment (i.e., by applying the z-score).
endpoint	A Boolean showing whether the endpoint adjustment mentioned in Agresti (1999) should be used. The default option is FALSE, in which case, the endpoint adjustment will not be made to the confidence interval. If it is set to TRUE, the endpoint adjustment will be made to the confidence interval only; no adjustment will be made to the p-value or the test statistic.

<code>adjust</code>	A character string specifying the cell count adjustment: "Agresti" for the independent-smoothed adjustment of Agresti (1999), "Gart" for the Gart (1966) adjustment of adding 0.5 to each cell, "Fagerland" for the Fagerland and Newcombe (2012) adjustment of adding 0.25 to each cell, "Woolf" for no adjustment, corresponding to the original Woolf (1955) logit interval, and "Custom" for custom adjustment.
<code>adjust.custom</code>	A numeric vector of length 4 specifying the custom cell count adjustments. This is used only when "Custom" is chosen for in the <code>adjust</code> input argument.
<code>se.option</code>	A character string specifying the standard error option: "original" for the original (untransformed) version and "sinh" for the inverse hyperbolic sine transformation of Newcombe (2001).
<code>conf.level</code>	A numeric specifying the confidence level. It must be a single number between 0 and 1. The default option is 0.95 for the 95% confidence level.

Details

This function provides new ways of computing an approximate odds ratio confidence interval for the 2-by-2 contingency table. Specifically, it allows the users to use the t-distribution with the Welch-adjusted degrees of freedom. Some brief descriptions can be found in Clinton, Morgan, and Noguchi (2026) although similar confidence intervals for the difference between the two proportions can also be found in Pan (2002).

Value

An object of class "htest" with components:

statistic The value of the t- or z-test statistic. It returns NA if the estimate is of an indefinite form.

parameter The degrees of freedom of the approximate t-distribution of the test statistic. It returns Inf if "Asymptotic" is chosen for the `dist` input argument. It returns NA if the estimate is of an indefinite form.

p.value P-value. It returns NA if the estimate is of an indefinite form.

estimate Estimated odds ratio. If the estimate is of an indefinite form such as 0/0, it returns NA.

conf.int Confidence interval for the population odds ratio parameter. It returns a unit interval if the estimate is of an indefinite form.

null.value Population of the odds ratio parameter under the null hypothesis.

alternative Character string describing the alternative hypothesis.

method Character string indicating the method used.

data.name Character string giving the names of the data.

References

- Agresti, A. (1999). On logit confidence intervals for the odds ratio with small samples. *Biometrics*, 55(2), 597–602.
- Clinton, C., Morgan, M., and Noguchi, K. (2026). A Comparison of Odds Ratio Confidence Interval Methods Utilizing Welch's Adjustment. 2026 Joint Mathematics Meetings, Washington, D.C.
- Fagerland, M. W., and Newcombe, R. G. (2013). Confidence intervals for odds ratio and relative risk based on the inverse hyperbolic sine transformation. *Statistics in Medicine*, 32(16), 2823–2836.

Gart, J. J. (1966). Alternative analyses of contingency tables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 28(1), 164–179.

Newcombe, R. G. (2001). Logit confidence intervals and the inverse sinh transformation. *The American Statistician*, 55(3), 200–202.

Pan, W. (2002). Approximate confidence intervals for one proportion and difference of two proportions. *Computational Statistics and Data Analysis*, 40(1), 143–157.

Woolf, B. (1955). On estimating the relation between blood group and disease. *Annals of Human Genetics*, 19(4), 251–253.

See Also

See [OR_Compare](#) for comparing the performances of different odds ratio confidence intervals.

Examples

```
#Testing with the data from Dube et al. (2009)
test_Dube <- OR_Welch(
  x = as.numeric(Dube_2009[1, 2:3]),
  n = as.numeric(Dube_2009[3, 2:3]))

#Another example using simulated data
OR_Welch(x = c(12, 5), n = c(20, 18))
```

plot.ORCompareCounts	<i>Plot Summary of Approximate Odds Ratio Confidence Interval Rankings</i>
----------------------	--

Description

Plots objects of the class "ORCompareCounts" to summarize the the top-performing approximate odds ratio confidence interval methods.

Usage

```
## S3 method for class 'ORCompareCounts'
plot(x, ...)
```

Arguments

x	An object of class "ORCompareCounts".
...	Additional arguments.

Details

This function displays the counts (frequencies) of the top-performing methods graphically.

Value

No return value, called for plotting objects of the class "ORCompareCounts".

See Also

[ORCompare_Counts](#), [summary.ORCompareCounts](#), [print.ORCompareCounts](#), [barplot.ORCompareCounts](#).

Examples

```
#Testing with the data from Wei et al. (2015)
test_Wei <- OR_Compare(
  x = as.numeric(Wei_2015[1, 2:3]),
  n = as.numeric(Wei_2015[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
test_Wei_counts <- ORCompare_Counts(test_Wei)

oldpar <- par(mfrow = c(1,2))
par(mar = c(17, 4.1, 4.1, 2.1))
method_rank_group <- as.factor(paste(test_Wei_counts$Method,
test_Wei_counts$Rank_Group))
plot(test_Wei_counts, xlab = "", ylab = "Count",
      main = "Counts vs. Methods", type = "h",
      col = "blue", xaxt = "n", lwd = 50, lend = "butt")
axis(1, at = 1:nrow(test_Wei_counts),
      labels = method_rank_group, las = 2)
par(oldpar)

#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
sim_results_counts <- ORCompare_Counts(sim_results)

oldpar <- par(mfrow = c(1,2))
par(mar = c(17, 4.1, 4.1, 2.1))
method_rank_group <- as.factor(paste(sim_results_counts$Method,
sim_results_counts$Rank_Group))
plot(sim_results_counts, xlab = "", ylab = "Count",
```

```

    main = "Counts vs. Methods", type = "h",
    col = "blue", xaxt = "n", lwd = 50, lend="butt")
axis(1, at = 1:nrow(sim_results_counts),
     labels = method_rank_group, las = 2)
par(oldpar)

```

print.ORCompare

Summary Table of Odds Ratio Confidence Interval Rankings

Description

Summarizes the rankings of different approximate odds ratio confidence interval methods.

Usage

```

## S3 method for class 'ORCompare'
print(x, ...)

```

Arguments

`x` An object of class "ORCompare" from [OR_Compare](#).
`...` Additional arguments.

Details

This function displays the rankings of approximate methods according to different metrics used.

Value

An "ORCompare" class object.

See Also

[OR_Compare](#), [summary.ORCompare](#).

Examples

```

#Testing with the data from Doll and Hill (1950)
test_Doll_Hill <- OR_Compare(
  x = as.numeric(Doll_Hill_1950[1, 2:3]),
  n = as.numeric(Doll_Hill_1950[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)

```

```

)
print(test_Doll_Hill)

#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
print(sim_results)

```

print.ORCompareCounts *Summary Table of Odds Ratio Confidence Interval Rankings*

Description

Summarizes how often the top-performing methods appear in the ranking.

Usage

```
## S3 method for class 'ORCompareCounts'
print(x, ...)
```

Arguments

x An object of class "ORCompareCounts" from [ORCompare_Counts](#).
... Additional arguments.

Details

This function displays how many times the top-performing methods appeared according to different metric(s) used.

Value

An "ORCompareCounts" class object.

See Also

[ORCompare_Counts](#), [summary.ORCompareCounts](#), [plot.ORCompareCounts](#), [barplot.ORCompareCounts](#).

Examples

```
#Testing with the data from Rake et al. (2009)
test_Rake <- OR_Compare(
  x = as.numeric(Rake_2009[1, 2:3]),
  n = as.numeric(Rake_2009[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
test_Rake_counts <- ORCompare_Counts(test_Rake)
print(test_Rake_counts)

#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
sim_results_counts <- ORCompare_Counts(sim_results)
print(sim_results_counts)
```

print.ORCompareSort	<i>Summary Table of Sorted Odds Ratio Confidence Interval Rankings</i>
---------------------	--

Description

Summarizes the top-performing methods by sorting them according to the rankings for the chosen metric(s).

Usage

```
## S3 method for class 'ORCompareSort'
print(x, ...)
```

Arguments

x	An object of class "ORCompareSort" from ORCompare_Sort .
...	Additional arguments.

Details

This function displays the top-performing methods as lists according to different metric(s) used.

Value

An "ORCompareSort" class object.

See Also

[ORCompare_Sort](#), [summary.ORCompareSort](#).

Examples

```
#Testing with the data from Sando et al. (2008)
test_Sando <- OR_Compare(
  x = as.numeric(Sando_2008[1, 2:3]),
  n = as.numeric(Sando_2008[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
test_Sando_sort <- ORCompare_Sort(test_Sando)
print(test_Sando_sort)
```

```
#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
sim_results_sort <- ORCompare_Sort(sim_results)
print(sim_results_sort)
```

Description

A 2-by-2 contingency table from Rake et al. (2009) comparing mesothelioma status between individuals who worked as carpenters for more than 10 years before age 30 and individuals employed in non-industrial occupations.

Usage

Rake_2009

Format

A data frame with 3 rows and 4 variables:

Mesothelioma_Status Mesothelioma status ("Mesothelioma", "None", "Total").

Carpenter_10plus_Years_Before_30 Number of individuals who worked as carpenters for more than 10 years before age 30.

Non_Industrial Number of individuals employed in non-industrial occupations.

Total Row totals.

The contingency table is:

Mesothelioma Status	Carpenter >10 Years Before 30	Non-Industrial	Total
Mesothelioma	66	15	81
None	25	278	303
Total	91	293	384

Details

The estimated odds ratio (OR) is

$$OR \approx 48.93.$$

That is, individuals who worked as carpenters for more than 10 years before age 30 have approximately 48.93 times the odds of being in the mesothelioma group compared to individuals employed in non-industrial occupations. This indicates a strong association between long-term occupational carpentry work and mesothelioma, a cancer commonly associated with asbestos exposure.

The proportion of mesothelioma cases among individuals who worked as carpenters for more than 10 years before age 30 is approximately 0.8148.

Source

Rake, C., Gilham, C., Hatch, J., Darnton, A., Hodgson, J., and Peto, J. (2009). See [doi:10.1038/sj.bjc.6604879](https://doi.org/10.1038/sj.bjc.6604879).

References

Rake, C., Gilham, C., Hatch, J., Darnton, A., Hodgson, J., and Peto, J. (2009). Occupational, domestic and environmental mesothelioma risks in the British population: a case-control study. *British Journal of Cancer*, 100, 1175–1183.

Examples

```
data(Rake_2009)
Rake_2009
n11 <- as.numeric(Rake_2009[1,2])
n12 <- as.numeric(Rake_2009[1,3])
n21 <- as.numeric(Rake_2009[2,2])
n22 <- as.numeric(Rake_2009[2,3])
OR <- (n11 * n22) / (n12 * n21)
OR
```

Sando_2008	<i>APOE Genotype and Alzheimer's Disease Status</i>
------------	---

Description

A 2-by-2 contingency table from Sando et al. (2008) comparing Alzheimer's disease status between individuals with the APOE e4/e4 genotype and individuals with the APOE e3/e3 genotype.

Usage

```
Sando_2008
```

Format

A data frame with 3 rows and 4 variables:

Alzheimers_Disease_Status Alzheimer's disease status ("Alzheimer's Disease", "None", "Total").

APOE_e4_e4 Number of individuals with the APOE e4/e4 genotype.

APOE_e3_e3 Number of individuals with the APOE e3/e3 genotype.

Total Row totals.

Details

The estimated odds ratio (OR) is $OR \approx 12.95$.

That is, individuals with the APOE e4/e4 genotype have approximately 12.95 times the odds of being in the Alzheimer's disease group compared to individuals with the APOE e3/e3 genotype, indicating a strong association between the APOE e4/e4 genotype and Alzheimer's disease.

The proportion of Alzheimer's disease cases among individuals with the APOE e4/e4 genotype is approximately 0.3497.

Source

Sando, S. B., Melquist, S., Cannon, A., Hutton, M.L., Sletvold, O., Saltvedt, I., White, L. R., Lydersen, S., and Aasly, J. O. (2008). See [doi:10.1186/1471237789](https://doi.org/10.1186/1471237789).

References

Sando, S. B., Melquist, S., Cannon, A., Hutton, M.L., Sletvold, O., Saltvedt, I., White, L. R., Lydersen, S., and Aasly, J. O. (2008). APOE e4 lowers age at onset and is a high risk factor for Alzheimer's disease; A case control study from central Norway. *BMC Neurology*, 8(9), 7 pages.

Examples

```
data(Sando_2008)
Sando_2008
n11 <- as.numeric(Sando_2008[1,2])
n12 <- as.numeric(Sando_2008[1,3])
n21 <- as.numeric(Sando_2008[2,2])
n22 <- as.numeric(Sando_2008[2,3])
OR <- (n11 * n22) / (n12 * n21)
OR
```

summary.ORCompare	<i>Summary Table of Odds Ratio Confidence Interval Rankings</i>
-------------------	---

Description

Summarizes the rankings of different approximate odds ratio confidence interval methods.

Usage

```
## S3 method for class 'ORCompare'
summary(object, ...)
```

Arguments

object	An object of class "ORCompare" from OR_Compare .
...	Additional arguments.

Details

This function displays the rankings of approximate methods according to different metrics used.

Value

An "ORCompare" class object.

See Also

[OR_Compare](#), [print.ORCompare](#).

Examples

```
#Testing with the data from Dube et al. (2009)
test_Dube <- OR_Compare(
  x = as.numeric(Dube_2009[1, 2:3]),
  n = as.numeric(Dube_2009[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
summary(test_Dube)
```

```
#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
summary(sim_results)
```

summary.ORCompareCounts

Summary Table of Odds Ratio Confidence Interval Rankings

Description

Summarizes how often the top-performing methods appear in the ranking.

Usage

```
## S3 method for class 'ORCompareCounts'
summary(object, ...)
```

Arguments

object	An object of class "ORCompareCounts" from ORCompare_Counts .
...	Additional arguments.

Details

This function displays how many times the top-performing methods appeared according to different metric(s) used.

Value

An "ORCompareCounts" class object.

See Also

[ORCompare_Counts](#), [print.ORCompareCounts](#), [plot.ORCompareCounts](#), [barplot.ORCompareCounts](#).

Examples

```
#Testing with the data from Wei et al. (2015)
test_Wei <- OR_Compare(
  x = as.numeric(Wei_2015[1, 2:3]),
  n = as.numeric(Wei_2015[3, 2:3]),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
test_Wei_counts <- ORCompare_Counts(test_Wei)
summary(test_Wei_counts)

#Another example using simulated data
sim_results <- OR_Compare(
  x = c(10, 20),
  n = c(30, 50),
  alternative = "two.sided",
  data.gen = "Binomial",
  conf.level = 0.95,
  nsim = 1000,
  fix.seed = TRUE,
  seed.num = 1234
)
sim_results_counts <- ORCompare_Counts(sim_results)
summary(sim_results_counts)
```

Description

Summarizes the top-performing methods by sorting them according to the rankings for the chosen metric(s).

Usage

```
## S3 method for class 'ORCompareSort'  
summary(object, ...)
```

Arguments

object	An object of class "ORCompareSort" from ORCompare_Sort .
...	Additional arguments.

Details

This function displays the top-performing methods as lists according to different metric(s) used.

Value

An "ORCompareSort" class object.

See Also

[ORCompare_Sort](#), [print.ORCompareSort](#).

Examples

```
#Testing with the data from Doll and Hill (1950)  
test_Doll_Hill <- OR_Compare(  
  x = as.numeric(Doll_Hill_1950[1, 2:3]),  
  n = as.numeric(Doll_Hill_1950[3, 2:3]),  
  alternative = "two.sided",  
  data.gen = "Binomial",  
  conf.level = 0.95,  
  nsim = 1000,  
  fix.seed = TRUE,  
  seed.num = 1234  
)  
test_Doll_Hill_sort <- ORCompare_Sort(test_Doll_Hill)  
summary(test_Doll_Hill_sort)
```

```
#Another example using simulated data  
sim_results <- OR_Compare(  
  x = c(10, 20),  
  n = c(30, 50),  
  alternative = "two.sided",  
  data.gen = "Binomial",  
  conf.level = 0.95,  
  nsim = 1000,
```

```

    fix.seed = TRUE,
    seed.num = 1234
  )
  sim_results_sort <- ORCompare_Sort(sim_results)
  summary(sim_results_sort)

```

Wei_2015

HLA-B27 Status and Ankylosing Spondylitis

Description

A 2-by-2 contingency table from Wei et al. (2015) comparing ankylosing spondylitis status between individuals who are HLA-B27 positive and individuals who are HLA-B27 negative.

Usage

```
Wei_2015
```

Format

A data frame with 3 rows and 4 variables:

Ankylosing_Spondylitis_Status Disease status ("Ankylosing Spondylitis", "None", "Total").

HLA_B27_Positive Number of HLA-B27 positive individuals.

HLA_B27_Negative Number of HLA-B27 negative individuals.

Total Row totals.

The contingency table is:

Ankylosing Spondylitis Status	HLA-B27+	HLA-B27-	Total
Ankylosing Spondylitis	431	40	471
None	43	514	557
Total	474	554	1028

Details

The estimated odds ratio (OR) is

$$OR \approx 128.80.$$

That is, HLA-B27 positive individuals have approximately 128.80 times the odds of having ankylosing spondylitis compared to HLA-B27 negative individuals, indicating an extremely strong association between HLA-B27 positivity and ankylosing spondylitis.

The proportion of ankylosing spondylitis cases among HLA-B27 positive individuals is approximately 0.9151.

Source

Wei, J. C.-C., Sung-Ching, H. W., Hsu, Y.-W., Wen, Y.-F., Wang, W.-C., Wong, R.-H., Lu, H.-F., van Gaalen, F. A., and Chang, W.C-. (2015). See [doi:10.1371/journal.pone.0137189](https://doi.org/10.1371/journal.pone.0137189).

References

Wei, J. C.-C., Wong, H. S.-C., Hsu, Y.-W., Wen, Y.-F., Wang, W.-C., Wong, R.-H., Lu, H.-F., van Gaalen, F. A., and Chang, W.C-. (2015). Interaction between HLA-B60 and HLA-B27 as a better predictor of ankylosing spondylitis in a Taiwanese population *PLoS ONE*, 10(10): e0137189.

Examples

```
data(Wei_2015)
Wei_2015
n11 <- as.numeric(Wei_2015[1,2])
n12 <- as.numeric(Wei_2015[1,3])
n21 <- as.numeric(Wei_2015[2,2])
n22 <- as.numeric(Wei_2015[2,3])
OR <- (n11 * n22) / (n12 * n21)
OR
```

Index

* datasets

Doll_Hill_1950, [4](#)

Dube_2009, [5](#)

Rake_2009, [19](#)

Sando_2008, [21](#)

Wei_2015, [26](#)

barplot.ORCompareCounts, [2](#), [7](#), [15](#), [17](#), [24](#)

Doll_Hill_1950, [4](#)

Dube_2009, [5](#)

OR_Compare, [2](#), [7](#), [8](#), [9](#), [14](#), [16](#), [22](#)

OR_Welch, [10](#), [11](#), [12](#)

ORCompare_Counts, [6](#), [11](#), [15](#), [17](#), [23](#), [24](#)

ORCompare_Sort, [8](#), [11](#), [18](#), [19](#), [25](#)

plot.ORCompareCounts, [7](#), [14](#), [17](#), [24](#)

print.ORCompare, [2](#), [16](#), [22](#)

print.ORCompareCounts, [7](#), [15](#), [17](#), [24](#)

print.ORCompareSort, [8](#), [18](#), [25](#)

Rake_2009, [19](#)

Sando_2008, [21](#)

summary.ORCompare, [2](#), [16](#), [22](#)

summary.ORCompareCounts, [7](#), [15](#), [17](#), [23](#)

summary.ORCompareSort, [8](#), [19](#), [24](#)

Wei_2015, [26](#)