Package ‘risks’

June 13, 2023

Title  Estimate Risk Ratios and Risk Differences using Regression
Version  0.4.2
Description  Risk ratios and risk differences are estimated using regression models that allow for binary, categorical, and continuous exposures and confounders. Implemented are marginal standardization after fitting logistic models (g-computation) with delta-method and bootstrap standard errors, Miettinen’s case-duplication approach (Schouten et al. 1993, \(<\text{doi}:10.1002/sim.4780121808>\)), log-binomial (Poisson) models with empirical variance (Zou 2004, \(<\text{doi}:10.1093/aje/kwh090>\)), binomial models with starting values from Poisson models (Spiegelman and Hertzmark 2005, \(<\text{doi}:10.1093/aje/kwi188>\)), and others.
License  GPL-3
Encoding  UTF-8
LazyData  true
RoxygenNote  7.2.3
URL  https://stopsack.github.io/risks/

BugReports  https://github.com/stopsack/risks/issues
Suggests  addreg, covr, knitr, logbin, MASS, rmarkdown, testthat
Imports  boot, bcaboot, broom, dplyr, purrr, rlang, sandwich, stats, tibble, tidyr
VignetteBuilder  knitr
Depends  R (>= 3.5.0)
NeedsCompilation  no
Author  Konrad Stopsack [aut, cre] (<https://orcid.org/0000-0002-0722-1311>), Travis Gerke [aut] (<https://orcid.org/0000-0002-9500-8907>)
Maintainer  Konrad Stopsack <stopsack@post.harvard.edu>
Repository  CRAN
Date/Publication  2023-06-13 08:10:02 UTC
### Description

A cohort of women with breast cancer and complete follow-up, as used by Spiegelman and Hertzmark (Am J Epidemiol 2005) and Greenland (Am J Epidemiol 2004).

### Usage

```r
breastcancer
```

### Format

#### breastcancer:

A tibble with 192 rows and 3 columns:

- **death** Death, binary: 0, 1
- **stage** Cancer stage, 3 categories
- **receptor** Hormone receptor status, binary: "High", "Low" ...

### Source

confint.duplicate  

Clustering-corrected confidence intervals for case duplication model

Description

Estimate confidence intervals for the case duplication model with robust/sandwich/empirical co-variance structure.

Usage

```r
## S3 method for class 'duplicate'
confint(object, parm = NULL, level = 0.95, ...)
```

Arguments

- `object`: Fitted model
- `parm`: Not used
- `level`: Confidence level, defaults to 0.95
- `...`: Additional arguments, not used

Value

Matrix: First column, lower bound; second column, upper bound.

confint.margstd_boot  

Bootstrap confidence intervals

Description

Confidence intervals for models fit using marginal standardization based on parametric bootstrapping.

Usage

```r
## S3 method for class 'margstd_boot'
confint(  
  object,  
  parm = NULL,  
  level = 0.95,  
  bootrepeats = 1000,  
  bootci = c("bca", "normal", "nonpar"),  
  jacksd = FALSE,  
  ...  
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>Model fitted through marginal standardization</td>
</tr>
<tr>
<td>parm</td>
<td>Not used, for compatibility</td>
</tr>
<tr>
<td>level</td>
<td>Confidence level, defaults to 0.95.</td>
</tr>
<tr>
<td>bootrepeats</td>
<td>Bootstrap repeats. Defaults to 1000. Consider increasing.</td>
</tr>
<tr>
<td>bootci</td>
<td>Type of bootstrap confidence interval:</td>
</tr>
<tr>
<td></td>
<td>• &quot;bca&quot; Default. Parametric BCa (bias-corrected accelerated) confidence intervals.</td>
</tr>
<tr>
<td></td>
<td>• &quot;normal&quot; Parametric normality-based confidence intervals, which require lower repeat numbers but are less accurate and may result in invalid results for ratios.</td>
</tr>
<tr>
<td></td>
<td>• &quot;nonpar&quot; Non-parametric BCa confidence intervals, which should be used with caution because of the risk of sparse-data bias with non-parametric bootstrapping.</td>
</tr>
<tr>
<td>jacksd</td>
<td>Return jackknife Monte-Carlo error for the confidence limits? Only functional with BCa confidence intervals. Defaults to FALSE.</td>
</tr>
<tr>
<td>...</td>
<td>Not used</td>
</tr>
</tbody>
</table>

Value

Matrix: First column, lower bound; second column, upper bound.

Description

Confidence intervals for models fit using marginal standardization based on the melta method.

Usage

```r
# S3 method for class 'margstd_delta'
confint(object, parm = NULL, level = 0.95, ...)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>Model fitted through marginal standardization, delta method</td>
</tr>
<tr>
<td>parm</td>
<td>Not used, for compatibility</td>
</tr>
<tr>
<td>level</td>
<td>Confidence level, defaults to 0.95.</td>
</tr>
<tr>
<td>...</td>
<td>Not used</td>
</tr>
</tbody>
</table>

Value

Matrix: First column, lower bound; second column, upper bound.
confint.robpoisson

**Robust confidence intervals for Poisson model**

**Description**

Estimate confidence intervals for the Poisson model with robust/sandwich/empirical covariance structure.

**Usage**

```r
# S3 method for class 'robpoisson'
confint(object, parm = NULL, level = 0.95, ...)
```

**Arguments**

- `object`: Fitted model
- `parm`: Not used
- `level`: Confidence level, defaults to 0.95
- `...`: Additional arguments, not used

**Value**

Matrix: First column, lower bound; second column, upper bound.

---

print.risks

**Print model**

**Description**

Print fitted risks model. The only change, compared to `print.glm()`, is the addition of the main type of model: relative risk or risk difference. If multiple models were fitted via `approach = "all"`, then the first converged model will be printed.

**Usage**

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

**Arguments**

- `x`: Fitted model
- `...`: Passed to `print.glm()`

**Value**

No return value, called for printing
print.summary.risks  Print model summary

Description

Print summaries for "risks" models. The printout is the same as for regular summaries of generalized linear models fit via stats::glm(), except that the type of "risks" model is printed first (e.g., "Poisson model with robust covariance") and confidence intervals for model parameters are printed at the end.

Usage

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

Arguments

x  Model
...
Passed on

Value

No return value, called for printing

riskratio  Fit risk ratio and risk difference models

Description

riskratio and riskdiff provide a flexible interface to fitting risk ratio and risk difference models. In cohort studies with a binary outcome, risk ratios and risk differences are typically more appropriate to report than odds ratios from logistic regression, yet such models have historically been difficult to implement in standard software.

The risks package selects an efficient way to fit risk ratio or risk difference models successfully, which will converge whenever logistic models converge. Optionally, a specific approach to model fitting can also be requested. Implemented are Poisson models with robust covariance, binomial models, logistic models with case duplication, binomial models aided in convergence by starting values obtained through Poisson models or logistic models with case duplication, binomial models fitted via combinatorial expectation maximization (optionally also with Poisson starting values), and estimates obtained via marginal standardization after logistic regression with bootstrapped or delta method for confidence intervals.

Adjusting for covariates (e.g., confounders) in the model specification (formula =) is possible.
riskratio

Usage

riskratio(
  formula,
  data,
  approach = c("auto", "all", "robpoisson", "duplicate", "glm", "glm_startp",
    "glm_startd", "glm_cem", "glm_cem_startp", "margstd_boot", "margstd_delta",
    "logistic", "legacy"),
  variable = NULL,
  at = NULL,
  ...
)

riskdiff(
  formula,
  data,
  approach = c("auto", "all", "robpoisson", "glm", "glm_startp", "glm_cem",
    "glm_cem_startp", "margstd_boot", "margstd_delta", "legacy"),
  variable = NULL,
  at = NULL,
  ...
)

Arguments

  formula  A formula object of the form response ~ predictors.
  data     A tibble or data.frame object.
  approach Optional: Method for model fitting.
    • "auto" (default) is recommended; it will return results of "margstd_delta"
      unless interaction terms between exposure and confounders are included.
      This these cases, results from "margstd_boot" are returned.
    • "all" will attempt to fit the model via all implemented approaches to allow
      for comparisons.
    • "legacy" selects the most efficient approach that converges and ensures
      that predicted probabilities are within range (< 1).
      The other options allow for directly selecting a fitting approach, some of which
      may not converge or yield out-of-range predicted probabilities. See full
documentation for details.
    • "glm" Binomial model.
    • "glm_startp" Binomial model with starting values from Poisson model.
    • "glm_startd" Binomial model with starting values from logistic model
      with case duplication.
    • "robpoisson" Poisson model with robust covariance.
    • "duplicate" Logistic model with duplication of cases. Only available in
      riskratio().
    • "glm_cem" Binomial model fitted with combinatorial expectation max-
      imization.
riskratio

- "glm_cem_startp" As glm_cem, with Poisson starting values.
- "margstd_boot" Marginal standardization after logistic model, bootstrap standard errors/confidence intervals.
- "margstd_delta" Marginal standardization after logistic model, delta method standard errors/confidence intervals.
- "logistic" For comparison only: the logistic model. Only available in riskratio().

variable Optional: exposure variable to use for marginal standardization. If variable is not provided and marginal standardization is attempted, then the first variable in the model is used as the exposure. Levels are determined automatically for variables types logical, character, factor and can optionally be supplied via `at =`.

at Optional: Levels of exposure variable `variable` for marginal standardization. `at =` determines the levels at which contrasts of the exposure are to be assessed. The level listed first is used as the reference. Levels must exist in the data for character, factor or ordered factor variables. For numeric variables, levels that do not exist in the data can be interpolations or extrapolations; if levels exceed the extremes of the data (extrapolation), a warning will be displayed.

Value

Fitted model. This object can be passed on to post-processing functions:

- `tidy.risks`: a tibble of coefficients and confidence intervals.

Standard post-processing functions can also be used:

- `coef`: a vector of coefficients.
- `confint`: a matrix of confidence intervals (risks-specific S3 methods: `confint.robpoisson`, `confint.margstd_boot`, `confint.margstd_delta`).
- `predict.glm(type = "response")`: fitted values (predictions).
- `residuals`: residuals.

If model fitting using all possible approaches was requested via `approach = "all"`, then their results can be retrieved from the list `all_models` in the returned object (e.g., `fit$all_models[[1]]`, `fit$all_models[[2]]`, etc.).

Functions

- `riskratio()`: Fit risk ratio models
- `riskdiff()`: Fit risk difference models
References


Spiegelman D, Hertzmark E. Easy SAS Calculations for Risk or Prevalence Ratios and Differences. Am J Epidemiol 2005;162:199-200. (Binomial models fitted using starting values from Poisson models; approach = "glm_start")


Donoghoe MW, Marschner IC. Stable computational methods for additive binomial models with application to adjusted risk differences. Comput Stat Data Anal 2014;80:184-96. (Additive binomial models fitted via combinatorial expectation maximization; riskdiff(approach = "glm_cem"))

Localio AR, Margolis DJ, Berlin JA. Relative risks and confidence intervals were easily computed indirectly from multivariable logistic regression. J Clin Epidemiol 2007;60(9):874-82. (Marginal standardization after fitting a logistic model; approach = "margstd_boot")

Examples

data(breastcancer) # Cohort study with binary outcome
# See for details: help(breastcancer)

# Risk ratio model
fit_rr <- riskratio(formula = death ~ stage + receptor, data = breastcancer)
fit_rr
summary(fit_rr)

# Risk difference model
fit_rd <- riskdiff(formula = death ~ stage + receptor, data = breastcancer)
fit_rd
summary(fit_rd)

risks

risks: Estimate risk ratios and risk differences using regression

Description

The risks package allows for fitting risk ratio and risk difference models using regression.
**risks functions**

- **riskratio**: Fit risk ratio models.
- **riskdiff**: Fit risk difference models.
- **summary.risks**: Summarize fitted model.
- **tidy.risks**: Tibble (data frame) of parameters, coefficients, standard errors, confidence intervals.
- **confint.robpoisson**, **confint.duplicate**, **confint.margstd_boot**, **confint.margstd_delta**: Confidence intervals. (Standard confidence intervals for generalized linear models are used for other models.)

**See Also**

https://github.com/stopsack/risks

---

```r
rr_rd_mantel_haenszel  Risk Ratios and Risk Differences from Mantel-Haenszel Estimators
```

**Description**

This function implements the Mantel-Haenszel estimators for risk ratio and risk differences for a binary or categorical exposure and one or more categorical confounder(s). Compare to estimates from regression models.

**Usage**

```r
rr_rd_mantel_haenszel(
  data,
  exposure,
  outcome,
  confounders,
  estimand = c("rr", "rd"),
  conf.level = 0.95
)
```

**Arguments**

- **data**
  Data set.
- **exposure**
  Exposure variable. Must be binary or categorical. The first level is treated as unexposed.
- **outcome**
  Outcome variable. Must be binary.
- **confounders**
  Optional. Binary or categorical variable(s) to perform stratification over. Supply more than one variable using confounders = c(var1, var2).
- **estimand**
  Optional. "rr" for risk ratio; "rd" for risk difference. Defaults to "rr".
- **conf.level**
  Optional. Confidence level. Defaults to 0.95.
Value

Tibble in tidy format with

- term: the (non-reference) exposure levels
- estimate: Risk ratio (on log scale) or risk difference
- std.error, conf.low, and conf.high: Square-root of M-H variance estimate, and the corresponding confidence limits (on log scale for RR)
- model: always "mh"
- estimand: "rr" or "rd"

References


Examples

# Newman SC. Biostatistical methods in epidemiology. New York, NY:
# Wiley, 2001, table 5.3

library(tibble) # used to set up example data
dat <- tibble(
  death = c(rep(1, 54), rep(0, 138)),
  stage = c(rep("Stage I", 7), rep("Stage II", 26), rep("Stage III", 21),
           rep("Stage I", 60), rep("Stage II", 70), rep("Stage III", 8)),
  receptor = c(rep("Low", 2), rep("High", 5), rep("Low", 9), rep("High", 17),
               rep("Low", 12), rep("High", 9), rep("Low", 10), rep("High", 50),
               rep("Low", 13), rep("High", 57), rep("Low", 2), rep("High", 6))

# Risk difference
rr_rd_mantel_haenszel(
  data = dat,
  exposure = stage,
  outcome = death,
  confounders = receptor,
  estimand = "rd")

# Risk ratio, log scale:
result <- rr_rd_mantel_haenszel(
  data = dat,
  exposure = stage,
  outcome = death,
  confounders = receptor,
  estimand = "rr")
result

# Risk ratio, exponentiated:
result %>%
dplyr::mutate(dplyr::across(.cols = c(estimate, conf.low, conf.high),
                           .fns = exp))
**summary.duplicate**  
Summary for logistic model with case duplication and cluster-robust covariance

**Description**

Summarize results from fitting a logistic model with case duplication and cluster-robust covariance. The output is the same as for a regular `summary(glm(...))`, except for using cluster-robust standard errors.

**Usage**

```r
## S3 method for class 'duplicate'
summary(
  object,
  dispersion = NULL,
  correlation = FALSE,
  symbolic.cor = FALSE,
  ...
)
```

**Arguments**

- `object`: Model
- `dispersion`: Not used
- `correlation`: Not used
- `symbolic.cor`: Not used
- `...`: Other arguments, not used

**Value**

Model summary (list)

**summary.margstd_boot**  
Summary for models using marginal standardization

**Description**

Summary for models using marginal standardization
Usage

## S3 method for class 'margstd_boot'

\textbf{summary(}
\begin{itemize}
  \item \texttt{object},\texttt{dispersion = NULL,}
  \item \texttt{correlation = FALSE,}
  \item \texttt{symbolic.cor = FALSE,}
  \item \texttt{level = 0.95,}
  \item \texttt{bootrepeats = 1000,}
  \item \texttt{bootci = c("bca", "normal", "nonpar"),}
\end{itemize}
\textbf{\ldots})

Arguments

- \textbf{object} \hspace{1cm} Model
- \textbf{dispersion} \hspace{1cm} Not used
- \textbf{correlation} \hspace{1cm} Not used
- \textbf{symbolic.cor} \hspace{1cm} Not used
- \textbf{level} \hspace{1cm} Confidence level, defaults to 0.95.
- \textbf{bootrepeats} \hspace{1cm} Bootstrap repeats for standard errors. Defaults to 1000. Consider increasing.
- \textbf{bootci} \hspace{1cm} Type of bootstrap confidence interval:
  - "bca" Default. Parametric BCa (bias-corrected accelerated) confidence intervals.
  - "normal" Parametric normality-based confidence intervals, which require lower repeat numbers but are less accurate and may result in invalid results for ratios.
  - "nonpar" Non-parametric BCa confidence intervals, which should be used with caution because of the risk of sparse-data bias with non-parametric bootstrapping.

\textbf{\ldots} \hspace{1cm} Not used

Value

Model summary (list)

---

\textbf{summary.margstd_delta} \hspace{1cm} \textit{Summary for models using marginal standardization with delta method SEs}

---

Description

Summary for models using marginal standardization with delta method SEs
## Summary.Risks

### Usage

```r
## S3 method for class 'margstd_delta'
summary(
  object,
  dispersion = NULL,
  correlation = FALSE,
  symbolic.cor = FALSE,
  level = 0.95,
  ...
)
```

### Arguments

- `object`: Model
- `dispersion`: Not used
- `correlation`: Not used
- `symbolic.cor`: Not used
- `level`: Confidence level, defaults to 0.95.
- `...`: Not used

### Value

Model summary (list)

----

### Description

Determine type of risks model fitted and generate appropriate summary.

#### Usage

```r
## S3 method for class 'risks'
summary(object, conf.int = TRUE, default = TRUE, ...)
```

#### Arguments

- `object`: Fitted model
- `conf.int`: Add confidence intervals to printout? Defaults to TRUE.
- `default`: Normal confidence intervals via `confint.default()`? Default to TRUE. By setting `default = FALSE`, profiling-based confidence intervals can be calculated for binomial models.
- `...`: Passed on
summary.robpoisson

Details

If multiple models were fitted (approach = "all"), then the first converged model is displayed. Other models can be accessed via the returned list $all_models$.

Value

Model summary (list)

summary.robpoisson  Summary for Poisson model with robust covariance

Description

Summarize results from fitting a Poisson model with robust/empirical/sandwich covariance. The output is the same as for a regular summary(glm(...)), except for using robust standard errors.

Usage

## S3 method for class 'robpoisson'
summary(
  object,
  dispersion = NULL,
  correlation = FALSE,
  symbolic.cor = FALSE,
  ...
)

Arguments

object  Model
dispersion  Not used
correlation  Not used
symbolic.cor  Not used
...  Other arguments, not used

Value

Model summary (list)
Description

Obtain a tibble (data frame) with parameters, coefficients, standard errors, confidence limits, and p-values. A column with the type of model fitted is added.

Usage

```r
## S3 method for class 'risks'
tidy(
x, conf.int = TRUE, conf.level = 0.95, bootrepeats = 1000, bootci = c("bca", "normal", "nonpar"), bootverbose = FALSE, exponentiate = FALSE, default = TRUE, 
...)
```

Arguments

- **x** Model
- **conf.int** Show confidence intervals? Optional. Confidence level. Defaults to 0.95.
- **conf.level** Optional. Number of bootstrap repeats. Applicable to models fitted via marginal standardization and bootstrapping (approach = "margstd_boot"). Defaults to 1000. Strongly recommended to increase repeats to »1000.
- **bootrepeats** Optional. Type of bootstrap confidence interval:
  - "bca" Default. Parametric BCa (bias-corrected accelerated) confidence intervals.
  - "normal" Parametric normality-based confidence intervals, which require lower repeat numbers but are less accurate and may result in invalid results for ratios.
  - "nonpar" Non-parametric BCa confidence intervals, which should be used with caution because of the risk of sparse-data bias with non-parametric bootstrapping.
- **bootverbose** Optional. Add values of bootrepeats and bootci parameters and the jackknife-based Monte-Carlo error for the confidence limits (only for type = "bca") to the returned tibble? Defaults to FALSE.
tidy.risks

`tidy.risks` 17

**exponentiate**  Optional. Exponentiate coefficients and confidence limits? Defaults to FALSE. Setting exponentiate = TRUE is useful for relative risk models (log links).

**default**  Use default, normality-based confidence intervals? Defaults to TRUE. With default = FALSE, for binomial models only, profile likelihood-based confidence intervals can be calculated.

...  Passed on

**Details**

If multiple types of models are fitted, tidy() can be used to parameters for all models at once, in one tibble. The last column of the tibble includes the name of the model. See examples.

**Value**

tibble

**Examples**

```r
# Define example data
library(broom)  # provides tidy() function
dat <- tibble::tibble(
  death = c(rep(1, 54), rep(0, 138)),
  stage = c(rep("Stage I", 7), rep("Stage II", 26), rep("Stage III", 21),
            rep("Stage I", 60), rep("Stage II", 70), rep("Stage III", 8)),
  receptor = c(rep("Low", 2), rep("High", 5), rep("Low", 9), rep("High", 17),
              rep("Low", 12), rep("High", 9), rep("Low", 10), rep("High", 50),
              rep("Low", 13), rep("High", 57), rep("Low", 2), rep("High", 6)))

# Fit and tidy the model
fit_rr <- riskratio(formula = death ~ stage + receptor, data = dat)
tidy(fit_rr)

# Marginal standardization,
# increase number of bootstrap repeats:
fit_rr <- riskratio(
  formula = death ~ stage + receptor, data = dat,
  approach = "margstd_boot")
tidy(fit_rr, bootrepeats = 2000)

# Multiple types of models fitted:
fit_rr <- riskratio(formula = death ~ stage + receptor, data = dat,
                     approach = "all")
tidy(fit_rr)
```
Index

* datasets
  breastcancer, 2

breastcancer, 2

c coef, 8
confint, 8
confint.duplicate, 3, 10
confint.margstd.boot, 3, 8, 10
confint.margstd_delta, 4, 8, 10
confint.robpoisson, 5, 8, 10

predict.glm, 8
print.risks, 5
print.summary.risks, 6

residuals, 8
riskdiff, 10
riskdiff(riskratio), 6
riskratio, 6, 10
risks, 9
rr_rd_mantel_haenszel, 10

summary.duplicate, 12
summary.margstd.boot, 8, 12
summary.margstd_delta, 8, 13
summary.risks, 8, 10, 14
summary.robpoisson, 8, 15

tidy, 11
 tidy.risks, 8, 10, 16