Package ‘fragilityindex’

July 17, 2017

Type Package
Title Fragility Index
Version 0.1.0
Description Implements and extends the fragility index calculation for dichotomous results as described in Walsh, Srinathan, McAuley, Mrkobrada, Levine, Ribic, Molnar, Dattani, Burke, Guyatt, Thabane, Walter, Pogue, and Devereaux (2014) <DOI:10.1016/j.jclinepi.2013.10.019>.
License MIT + file LICENSE
Imports base, stats, pbapply, survival, stringr
LazyData TRUE
RoxygenNote 6.0.1
NeedsCompilation no
Author Kipp Johnson [aut, cre]
Maintainer Kipp Johnson <kipp.william.johnson@gmail.com>
Repository CRAN
Date/Publication 2017-07-17 21:36:08 UTC

R topics documented:

fragility.index .................................................. 2
logisticfragility ............................................... 3
revfragility.index ............................................ 4
survivalfragility .............................................. 5

Index

1
fragility.index  Fragility Index Calculation

Description

Compute the fragility index for a dichotomous outcome, i.e. the number of flipped outcomes between cases and control it would take to make a significant-result non-significant.

Usage

`fragility.index(intervention_event, control_event, intervention_n, control_n, conf.level = 0.95, verbose = FALSE, print.mat = FALSE)`

Arguments

- `intervention_event` Number of events in intervention group
- `control_event` Number of events in control group
- `intervention_n` Total number of patients in intervention group
- `control_n` Total number of patients in the control group
- `conf.level` Significance level
- `verbose` Logical indicating if function will return verbose results or only fragility index
- `print.mat` Logical indicating if 2x2 matrices should be printed for each iteration of algorithm

Value

If `verbose` is `FALSE`, returns a list with fragility index. If `verbose` is `TRUE`, returns a list with p-values for each fragility index at each iteration of the algorithm.

Examples

`fragility.index(15, 5, 40, 40)`
logisticfragility

Logistic Fragility Function

Description

Compute the fragility of a coefficient in a logistic regression for dichotomous outcomes, i.e. the number of removed observations it would take to make a significant-result non-significant. Uses the glm() function from the stats package.

Usage

logisticfragility(formula, data, covariate = "all.factors.default", conf.level = 0.95, verbose = FALSE)

Arguments

- **formula**: Model formula which will be evaluated by glm()
- **data**: Dataframe with values for model forma, passed to glm()
- **covariate**: Vector of covariates to find fragility index for. Default is all covariates in formula
- **conf.level**: Significance level
- **verbose**: Logical indicating if function will return verbose results or only fragility index

Value

If verbose is FALSE, returns a list with fragility indices for selected covariates. If verbose is TRUE, returns a list with p-values for each fragility index at each iteration of the algorithm.

Examples

```r
# Import and format example data
mydata <- read.csv("https://stats.idre.ucla.edu/stat/data/binary.csv")
mydata$rank <- factor(mydata$rank)

logisticfragility(admit ~ gre + gpa + rank, data = mydata, covariate="gpa", verbose = TRUE)
logisticfragility(admit ~ gre + gpa + rank, data = mydata)
```
revfragility.index  Reverse Fragility Index Calculation

Description

Compute the reverse fragility index for a dichotomous outcome, i.e. the number of flipped cases it would take to make a non-significant result significant.

Usage

revfragility.index(intervention_event, control_event, intervention_n, control_n, conf.level = 0.95, verbose = FALSE, print.mat = FALSE)

Arguments

- intervention_event: Number of events in intervention group
- control_event: Number of events in control group
- intervention_n: Total number of patients in intervention group
- control_n: Total number of patients in the control group
- conf.level: Significance level
- verbose: Logical indicating if function will return verbose results or only fragility index
- print.mat: Logical indicating if 2x2 matrices should be printed for each iteration of algorithm

Value

If verbose is FALSE, returns a list with fragility index. If verbose is TRUE, returns a list with p-values for each fragility index at each iteration of the algorithm.

Examples

revfragility.index(6,5,50,50, verbose=TRUE, print.mat=FALSE)
survivalfragility  Survival Fragility Function

Description
Compute the fragility of a coefficient in a survival test, i.e. the number of removed observations it would take to make a significant-result non-significant. Uses the coxph() function from the survival package.

Usage
survivalfragility(formula, data, covariate = "all.factors.default", conf.level = 0.95, verbose = FALSE)

Arguments
- formula: Model formula which will be evaluated by coxph()
- data: Dataframe with values for model forma, passed to coxph()
- covariate: Vector of covariates to find fragility index for. Default is all covariates in formula
- conf.level: Significance level
- verbose: Logical indicating if function will return verbose results or only fragility index

Value
If verbose is FALSE, returns a list with fragility indices for selected covariates. If verbose is TRUE, returns a list with p-values for each fragility index at each iteration of the algorithm.

Examples
library(survival); data <- lung
data$status = lung$status - 1 # recode status as a 0/1 variable

survivalfragility(Surv(time, status) ~ pat.karno + strata(inst),
data, covariate = "pat.karno")
survivalfragility(Surv(time, status) ~ pat.karno + ph.karno + strata(inst),
data, verbose = TRUE)
#algorithm does not converge for strata(inst)
survivalfragility(Surv(time, status) ~ pat.karno + ph.karno + strata(inst),
data, covariate = c("pat.karno","ph.karno"))
Index

fragility.index, 2
logisticfragility, 3
revfragility.index, 4
survivalfragility, 5