

Package ‘pARccs’

April 19, 2009

Type Package

Title Estimation of partial attributable risks (PAR) from case-control data

Version 0.1-1

Date 2008-03-03

Author Christiane Raemsch

Maintainer Christiane Raemsch <christiane_raemsch@yahoo.de>

Depends R (>= 2.4.1), e1071, cba

Description Functions to estimate partial attributable risks from case-control data with corresponding percentile or BCa confidence intervals

License GPL (>= 2)

Repository CRAN

Date/Publication 2008-10-15 16:25:13

R topics documented:

AdjAR	2
AR	4
Boot_CI	6
PAR	9

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

AdjAR	<i>Calling the estimation of the (adjusted) attributable risks from case-control data</i>
-------	---

Description

AdjAR realizes the estimation of (adjusted) attributable risk (AR) from case-control data via logistic regression by calling the adequate function which holds the computation

Usage

```
AdjAR(D, E, C = NULL, model)
```

Arguments

D	a vector which holds the case-control state ("1" = case, "0"=control)
E	a matrix of the exposure factor/s (all of them have to be dichotomous!)
C	a matrix of the confounder/s (all of them have to be categorical!)
model	a model formula or an object of class "glm"

Details

Depending from the entered data AdjAR accesses to two additional functions: AR_w0C is selected if there is no variable which is only a confounder, expressed as C=NULL. AR_wC is selected if there are also variables which only act as confounders, that means C is a matrix.

See [AR_w0C](#) and [AR_wC](#) for further information about the computation.

Value

AdjAR returns a matrix containing the attributable risk for every possible (binary) combination of the exposure factors in E.

If C=NULL these are only adjusted to the rest of the exposure factors (which are not part of the interested combination). If there are given confounders in C the attributable risks are additionally adjusted to them.

Note

Also if there are only a single exposure factor/confounder you have to enter a matrix, so this will be a matrix with only one column.

It is also important that the given variables in D, E and C are not defined as factors.

The names of the variables (outcome, exposure factor/s, confounder/s) in the argument model have to be identical to the (column-)names of the entered data. Furthermore all given exposure factors and confounders have to be part of the argument model.

Validity of the estimation can only be taken for granted for data with simple random sampling, stratified random sampling or frequency-matching of controls.

Author(s)

Christiane Raemsch

References

Levin, M. (1953) The occurrence of lung cancer in man *Acta Unio Internationalis Contra Cancrum* **9**, 531-41

Bruzzi, P.; Green S.; Byard, D. *et al.* (1985) Estimating the population attributable risk for multiple risk factors using case-control data *American Journal of Epidemiology* **122**, 904-14

Benichou, J. (1991) Methods of adjustment for estimating the attributable risk in case-control studies: a review *Statistics in Medicine* **10**, 1753-73

See Also

[AR_woC](#), [AR_wC](#)

Examples

```
##### Computation of the AR for every combination of two #####
##### exposure factors if there are no confounders #####

set.seed(2007)
dicho      <- c(0,1)
cc_state   <- sample(dicho, 100, replace=TRUE)
exposure1  <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2  <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
relation   <- as.formula(cc_state~exposure1+exposure2)
data_exp   <- cbind(exposure1, exposure2)
AR_exposures <- AdjAR(D=cc_state, E=data_exp, model=relation)

##### Computation of the AR for every combination of two #####
##### exposure factors with adjustment to confounder1 #####

set.seed(2008)
cc_state    <- sample(dicho, 100, replace=TRUE)
exposure1   <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2   <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
cat_confounder <- c(0,1,2,3)
confounder1 <- sample(cat_confounder, 100, replace=TRUE)
data_exp    <- cbind(exposure1, exposure2)
conf        <- matrix(confounder1, ncol=1)
colnames(conf) <- c("confounder1")
rel_mod     <- glm(cc_state~exposure1+exposure2+confounder1,
                  family=binomial)
AR_exposures <- AdjAR(cc_state, data_exp, conf, rel_mod)
```

Description

With the functions `AR_wC` and `AR_woC` the estimation of the attributable risks (AR) from case-control data is realized.

From `AR_woC` you get the ARs for the exposure factors of primary interest adjusted to the rest of the exposure factors, the resulting ARs for the exposure factors of primary interest from `AR_wC` are additionally adjusted to the given confounders.

Usage

```
AR_woC(D, E, model, bincomE, conf = NULL)
```

```
AR_wC(D, E, C = NULL, model, bincomE, conf = NULL)
```

Arguments

<code>D</code>	a vector which holds the case-control state ("1" = case, "0"=control)
<code>E</code>	a matrix of the exposure factor/s (all of them have to be dichotomous!)
<code>C</code>	a matrix of the confounder/s (all of them have to be categorical!)
<code>model</code>	a model formula or an object of class "glm"
<code>bincomE</code>	a matrix which contains all binary combinations of the exposures in <code>E</code>
<code>conf</code>	a vector which holds the corresponding number of column of the exposure factors which should act as an additional confounder; the default is <code>NULL</code> , which means no exposure factor acts as an additional confounder

Details

With `AR_wC` the (joint) attributable risk for the exposure factor(s) of primary interest, which are not mentioned in vector `conf`, is estimated. It is adjusted to the rest of exposure factor/s in `E` (these are defined by `conf`). If you want an additional adjustment to secondary confounders, use function `AR_woC` with indicating the confounder/s in `C`.

If `conf=NULL` the joint attributable risk of all given exposure factors is estimated (as the case may be with adjustment to the confounder/s).

For the adjusted estimation regression models are used, here it is a logistic regression model. Through this model the needed Odds Ratio (OR) is estimated. The argument `model` can be either of the form `D~terms`, where `terms` is a series of terms out of the exposure factors and confounders, or an object of class "glm". (In the process of model fitting with `glm` you have to choose `family=binomial` to get a logistic regression model.)

All given exposure factors and confounders have to be part of the argument `model`. The names of the variables (outcome, exposure factors, confounders) in the argument `model` have to be identical to the (col-)names of the entered data. Also the colnames of `bincomE` have to be identical to the colnames of `E`.

To get the matrix `bincomE` you may use the function `bincombinations()` (use `help(bincombinations, package=e1071)` for further information).

Value

`AR_woC` returns a single value which is the (joint) attributable risk of one (or more) exposure factor(s) adjusted to the rest of the exposure factors.

`AR_wC` returns a single value which is the (joint) attributable risk of one (or more) exposure factor(s) adjusted to the rest of the exposure factors and to the given confounders.

Note

Also if there are only a single exposure factor/confounder you have to enter a matrix, so this will be a matrix with only one column.

It is also important that the given variables in `D`, `E` and `C` are not defined as factors.

Validity of the estimation can only be taken for granted for data with simple random sampling, stratified random sampling or frequency-matching of controls.

Here the (adjusted) attributable risk for only one defined (binary) combination of the exposure factors is estimated. To get the (adjusted) attributable risks for every possible (binary) combination of all given exposure factors use function [AdjAR](#).

Author(s)

Christiane Raemsch

References

Levin, M. (1953) The occurrence of lung cancer in man *Acta Unio Internationalis Contra Cancrum* **9**, 531-41

Bruzzi, P.; Green S.; Byard, D. *et al.* (1985) Estimating the population attributable risk for multiple risk factors using case-control data *American Journal of Epidemiology* **122**, 904-14

Benichou, J. (1991) Methods of adjustment for estimating the attributable risk in case-control studies: a review *Statistics in Medicine* **10**, 1753-73

See Also

[AdjAR](#)

Examples

```
##### use of function 'AR_woC':          #####
##### attributable risk for exposure2    #####
##### adjusted for exposure 1           #####

set.seed(2007)
dicho      <- c(0,1)
cc_state   <- sample(dicho, 100, replace=TRUE)
exposure1  <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
```

```

exposure2      <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
relation       <- as.formula(cc_state~exposure1+exposure2)
data_exp      <- cbind(exposure1, exposure2)
bincom        <- bincombinations(2)
colnames(bincom) <- colnames(data_exp)
AR_exposure2  <- AR_wC(cc_state, data_exp, relation, bincom, c(2))

##### use of function 'AR_wC': #####
##### joint attributable risk for exposure1 #####
##### and exposure2 adjusted for confounder1 #####

set.seed(2008)
dicho         <- c(0,1)
cc_state      <- sample(dicho, 100, replace=TRUE)
exposure1     <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2     <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
cat_confounder <- c(0,1,2,3)
confounder1    <- sample(cat_confounder, 100, replace=TRUE)
rel_mod       <- glm(cc_state~exposure1+exposure2+confounder1,
                    family=binomial)
data_exp      <- cbind(exposure1, exposure2)
conf          <- matrix(confounder1, ncol=1)
colnames(conf) <- c("confounder1")
bincom        <- bincombinations(2)
colnames(bincom) <- colnames(data_exp)
AR_exposure1_2 <- AR_wC(cc_state, data_exp, conf, rel_mod, bincom)

```

Boot_CI	<i>Bootstrap confidence intervals for partial attributable risks (PAR) from case-control data</i>
---------	---

Description

With `Boot_CI` you can determine confidence intervals for partial attributable risks from case-control data. Therefor the nonparametric bootstrap is used with whose bootstrap replications either percentile confidence intervals or BCa confidence intervals are developed (or both, if you want to).

Usage

```

Boot_CI(D, E, C = NULL, model, stepwise = FALSE, scope = NULL,
        nboot = 1000, alpha = 0.025, original,
        type = c("perc", "bca", "both"), strat_boot = TRUE)

```

Arguments

D	a vector which holds the case-control state ("1" = case, "0"=control)
E	a matrix of the exposure factor/s (all of them have to be dichotomous!)
C	a matrix of the confounder/s (all of them have to be categorical!)

<code>model</code>	a model formula or an object of class "glm"
<code>stepwise</code>	a logical value indicating whether a stepwise-selected model should be used in the computation, default is <code>FALSE</code>
<code>scope</code>	a description of the variables which should be taken into account in the stepwise selection (upper model) and which variables are necessarily part of the model (lower model)
<code>nboot</code>	number of (bootstrap-)replication, default is 250
<code>alpha</code>	left- and right-hand error (default is 0.025), so you will get a $100 \cdot (1 - 2\alpha)\%$ confidence interval
<code>original</code>	a vector of the computed partial attributable risks from the original data
<code>type</code>	a description of the type of confidence intervals which should be computed, "perc" stands for the percentile confidence interval, "bca" for the BCa confidence interval. You should choose "both" if you want to have calculated both types of confidence intervals. <code>type="perc"</code> is the default.
<code>strat_boot</code>	a logical value indicating whether a stratified or a non-stratified bootstrap should be executed, default is <code>TRUE</code>

Details

The computation of the partial attributable risks from the data set does not take place in this function. You have to estimate them separately and pass the results through `original` to the function `Boot_CI`.

To generate the bootstrap sample in every replication step one may use either the stratified or the non-stratified method. If `strat_boot=TRUE` the sampling occurs separately from case-data and control-data, otherwise the sampling occurs from the complete data set.

If `stepwise=TRUE` the logistic regression model fitting the data from the bootstrap sample is chosen in a stepwise algorithm by the AIC. Therefore the argument `scope` is needed (look `?step` for more information). Note, that at least the main effects of the exposure factors (and confounders) have to be part of the lower model, so that the stepwise algorithm is only used to identify the most significant interactions. If `stepwise=FALSE` the formula of the argument `model` is used to build a model fitting the data of the bootstrap sample.

The bootstrap replications for the partial attributable risks are used to build confidence intervals (as default 95% confidence intervals are computed). Therefore two methods are implemented: the percentile method (`type="perc"`) and the bias-corrected and accelerated (BCa) method (`type="bca"`). In conjunction with the choice between these two methods you should take note of the great computational effort by using the BCa method.

Value

`Boot_CI` returns a named matrix with two columns: the first contains the lower endpoint, the second the upper endpoint.

Note

Also if there are only a single exposure factor/confounder you have to enter a matrix, so this will be a matrix with only one column.

The names of the variables (outcome, exposure factors, confounders) in the argument `model` have to be identically to the (column-)names of the entered data. Furthermore all given exposure factors and confounders have to be part of the argument `model`.

It is also important that the given variables in `D`, `E` and `C` are not defined as factors.

Validity of the interval estimation can only be taken for granted for data with simple random sampling, stratified random sampling or frequency-matching of controls.

Author(s)

Christiane Raemsch

References

Efron, B.; Tibshirani, R. (1986) Bootstrap methods for standard errors, confidence intervals, and other measure of statistical accuracy *Statistical Science* **1**, 54-75

Efron, B.; Tibshirani, R. (1993) *An Introduction to the Bootstrap* Chapman & Hall (Monographs on Statistics and Applied Probability 57)

See Also

[PAR](#)

Examples

```
##### Computation of BCa confidence intervals #####
##### for the PAR if there are no confounders #####

set.seed(2007)
dicho      <- c(0,1)
cc_state   <- sample(dicho, 100, replace=TRUE)
exposure1  <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2  <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
relation   <- as.formula(cc_state~exposure1+exposure2)
data_exp   <- cbind(exposure1, exposure2)
PAR_exposures <- PAR(cc_state, data_exp, model=relation)
CI_95      <- Boot_CI(D=cc_state, E=data_exp, model=relation,
                     nboot=70,original=PAR_exposures, type="bca")

##### Computation of percentile confidence intervals #####
##### for the PAR if there are confounders #####

set.seed(2008)
dicho      <- c(0,1)
cc_state   <- sample(dicho, 100, replace=TRUE)
exposure1  <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2  <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
cat_confounder <- c(0,1,2,3)
confounder1 <- sample(cat_confounder, 100, replace=TRUE)
relation   <- as.formula(cc_state~exposure1+exposure2+confounder1)
data_exp   <- cbind(exposure1, exposure2)
```

```
conf          <- matrix(confounder1, ncol=1)
colnames(conf) <- c("confounder1")
PAR_exposures <- PAR(cc_state, data_exp, conf, model=relation)
CI_95        <- Boot_CI(D=cc_state, E=data_exp, C=conf, model=relation,
                        nboot=70, original=PAR_exposures)
```

PAR

Estimating partial attributable risks from case-control data

Description

PAR estimates the partial attributable risks (PAR) for multiple exposure factors. The underlying data have to arise from a case-control-study.

Usage

```
PAR(D, E, C = NULL, model)
```

Arguments

D	a vector which holds the case-control state ("1" = case, "0"=control)
E	a matrix of the exposure factor/s (all of them have to be dichotomous!)
C	a matrix of the confounder/s (all of them have to be categorical!)
model	a model formula or an object of class "glm".

Details

For the estimation of the PAR the partitioning technique through the interstep of the sequential attributable risks by Eide and Gefeller (1995) is used.

It is assumed that all exposure factors are equally ranking and that there are no equally or hierarchically structured classes of exposure factors.

The needed (adjusted) attributable risks are estimated within the function with access to the function [AdjAR](#) (look there for further information).

Value

PAR returns a named matrix which contains the partial attributable risk for every given exposure factor.

Note

Also if there are only a single exposure factor/confounder you have to enter a matrix, so this will be a matrix with only one column.

It is also important that the given variables in D, E and C are not defined as factors.

The names of the variables (outcome, exposure factors, confounders) in the argument `model` have to be identical to the (column-)names of the entered data. Furthermore all given exposure factors and confounders have to be part of the argument `model`.

Validity of the estimation can only be taken for granted for data with simple random sampling, stratified random sampling or frequency-matching of controls.

To simplify the computation a compressed definition for the PAR is used (see Wille and Gefeller (1996) for detailed information).

Author(s)

Christiane Raemisch

References

Eide, G.; Gefeller, O. (1995) Sequential and average attributable fractions as aids in the selection of preventive strategies *Journal of Clinical Epidemiology* **48**, 645-55

Wille, L.; Gefeller, O. (1996) Partitioning the disease risk among several exposure factors: a computational solution to an epidemiological problem *Advances in Statistical Software* **5**, 249-56

Land, M.; Vogel, C.; Gefeller, O. (2001) Partitioning methods for multifactorial risk attribution *Statistical Methods in Medical Research* **10**, 217-30

See Also

[AdjAR](#)

Examples

```
#### partial attributable risks of exposure1      #####
#### and exposure2 if there are no confounders  #####

set.seed(2007)
dicho          <- c(0,1)
cc_state       <- sample(dicho, 100, replace=TRUE)
exposure1     <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2     <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
relation      <- as.formula(cc_state~exposure1+exposure2)
data_exp      <- cbind(exposure1, exposure2)
PAR_exposures <- PAR(D=cc_state, E=data_exp, model=relation)

#### partial attributable risks of exposure1 and  #####
#### exposure2 with taking into account confounder1 #####

set.seed(2008)
```

```
dicho <- c(0,1)
cc_state <- sample(dicho, 100, replace=TRUE)
exposure1 <- sample(dicho, 100, replace=TRUE, prob=c(0.7, 0.3))
exposure2 <- sample(dicho, 100, replace=TRUE, prob=c(0.4, 0.6))
cat_confounder <- c(0,1,2,3)
confounder1 <- sample(cat_confounder, 100, replace=TRUE)
data_exp <- cbind(exposure1, exposure2)
conf <- matrix(confounder1, ncol=1)
colnames(conf) <- c("confounder1")
rel_mod <- glm(cc_state~exposure1+exposure2+confounder1,
              family=binomial)
PAR_exposures <- PAR(cc_state, data_exp, conf, rel_mod)
```

Index

*Topic **manip**

AdjAR, 1
AR, 3
Boot_CI, 6
PAR, 9

AdjAR, 1, 5, 9, 10

AR, 3

AR_wC, 2, 3

AR_wC (AR), 3

AR_woC, 2, 3

AR_woC (AR), 3

Boot_CI, 6

PAR, 8, 9