Package 'AF'

October 12, 2022

Title Model-Based Estimation of Confounder-Adjusted Attributable Fractions

Version 0.1.5

Author Elisabeth Dahlqwist and Arvid Sjolander

Maintainer Elisabeth Dahlqwist <elisabeth.dahlqwist88@gmail.com>

Description Estimates the attributable fraction in different sampling designs adjusted for measured confounders using logistic regression (cross-sectional and case-control designs), conditional logistic regression (matched case-control design), Cox proportional hazard regression (cohort design with time-to-event outcome), gamma-frailty model with a Weibull baseline hazard and instrumental variables analysis. An exploration of the AF with a genetic exposure can be found in the package 'AFheritability' Dahlqwist E et al. (2019) <doi:10.1007/s00439-019-02006-8>.

License GPL-2 | GPL-3

Encoding UTF-8

LazyData true

Depends stats, base, graphics, survival, drgee, stdReg, R (>= 3.5.0), data.table, ivtools

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-05-20 18:00:10 UTC

R topics documented:

`AF.cc` *Attributable fraction for maches and non-matched case-control sampling designs. NOTE! Deprecated function. Use* `AFglm` *(for unmatched case-control studies) or* `AFclogit` *(for matched case-control studies).*

Description

`AF.cc` estimates the model-based adjusted attributable fraction for data from matched and non-matched case-control sampling designs.

Usage

```
AF.cc(formula, data, exposure, clusterid, matched = FALSE)
```

Arguments

`formula`	an object of class "`formula`" (or one that can be coerced to that class): a symbolic description of the model used for confounder adjustment. The exposure and confounders should be specified as independent (right-hand side) variables. The outcome should be specified as dependent (left-hand side) variable. The formula is used to object a logistic regression by `glm` for non-matched case-control and conditional logistic regression by `gee` (in package `drgee`) for matched case-control.
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in `data`, the variables are taken from environment (`formula`), typically the environment from which the function is called.
`exposure`	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
`clusterid`	the name of the cluster identifier variable as a string, if data are clustered (e.g. matched).
`matched`	a logical that specifies if the sampling design is matched (TRUE) or non-matched (FALSE) case-control. Default setting is non-matched (`matched = FALSE`).

Details

`Af.cc` estimates the attributable fraction for a binary outcome `Y` under the hypothetical scenario where a binary exposure `X` is eliminated from the population. The estimate is adjusted for confounders `Z` by logistic regression for unmatched case-control (`glm`) and conditional logistic regression for matched case-control (`gee`). The estimation assumes that the outcome is rare so that the risk ratio can be approximated by the odds ratio, for details see Bruzzi et. al. Let the AF be defined as

$$AF = 1 - \frac{Pr(Y_0 = 1)}{Pr(Y = 1)}$$

where $Pr(Y_0 = 1)$ denotes the counterfactual probability of the outcome if the exposure would have been eliminated from the population. If `Z` is sufficient for confounding control then the probability $Pr(Y_0 = 1)$ can be expressed as

$$Pr(Y_0 = 1) = E_Z\{Pr(Y = 1 \mid X = 0, Z)\}.$$

Using Bayes' theorem this implies that the AF can be expressed as

$$AF = 1 - \frac{E_Z\{Pr(Y = 1 \mid X = 0, Z)\}}{Pr(Y = 1)} = 1 - E_Z\{RR^{-X}(Z) \mid Y = 1\}$$

where $RR(Z)$ is the risk ratio

$$\frac{Pr(Y = 1 \mid X = 1, Z)}{Pr(Y = 1 \mid X = 0, Z)}.$$

Moreover, the risk ratio can be approximated by the odds ratio if the outcome is rare. Thus,

$$AF \approx 1 - E_Z\{OR^{-X}(Z) \mid Y = 1\}.$$

The odds ratio is estimated by logistic regression or conditional logistic regression. If `clusterid` is supplied, then a clustered sandwich formula is used in all variance calculations.

Value

`AF.est` estimated attributable fraction.

`AF.var` estimated variance of `AF.est`. The variance is obtained by combining the delta methods with the sandwich formula.

`log.or` a vector of the estimated log odds ratio for every individual. `log.or` contains the estimated coefficient for the exposure variable `X` for every level of the confounder `Z` as specified by the user in the formula. If the model to be estimated is

$$logit\{Pr(Y = 1|X, Z)\} = \alpha + \beta X + \gamma Z$$

then `log.or` is the estimate of β. If the model to be estimated is

$$logit\{Pr(Y = 1|X, Z)\} = \alpha + \beta X + \gamma Z + \psi XZ$$

then `log.odds` is the estimate of $\beta + \psi Z$.

`object` the fitted model. Fitted using logistic regression, `glm`, for non-matched case-control and conditional logistic regression, `gee`, for matched case-control.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Bruzzi, P., Green, S. B., Byar, D., Brinton, L. A., and Schairer, C. (1985). Estimating the population attributable risk for multiple risk factors using case-control data. *American Journal of Epidemiology* **122**, 904-914.

See Also

The new and more general version of the function: `AFglm` for non-matched and `AFclogit` for matched case-control sampling designs. `glm` and `gee` used for fitting the logistic regression model (for non-matched case-control) and the conditional logistic regression model (for matched case-control).

Examples

```
expit <- function(x) 1 / (1 + exp( - x))
NN <- 1000000
n <- 500

# Example 1: non matched case-control
# Simulate a sample from a non matched case-control sampling design
# Make the outcome a rare event by setting the intercept to -6
intercept <- -6
Z <- rnorm(n = NN)
X <- rbinom(n = NN, size = 1, prob = expit(Z))
Y <- rbinom(n = NN, size = 1, prob = expit(intercept + X + Z))
population <- data.frame(Z, X, Y)
Case <- which(population$Y == 1)
Control <- which(population$Y == 0)
# Sample cases and controls from the population
case <- sample(Case, n)
control <- sample(Control, n)
data <- population[c(case, control), ]

# Estimation of the attributable fraction
AF.cc_est <- AF.cc(formula = Y ~ X + Z + X * Z, data = data, exposure = "X")
summary(AF.cc_est)

# Example 2: matched case-control
# Duplicate observations in order to create a matched data sample
# Create an unobserved confounder U common for each pair of individuals
U  <- rnorm(n = NN)
Z1 <- rnorm(n = NN)
Z2 <- rnorm(n = NN)
X1 <- rbinom(n = NN, size = 1, prob = expit(U + Z1))
X2 <- rbinom(n = NN, size = 1, prob = expit(U + Z2))
Y1 <- rbinom(n = NN, size = 1, prob = expit(intercept + U + Z1 + X1))
Y2 <- rbinom(n = NN, size = 1, prob = expit(intercept + U + Z2 + X2))
```

```
# Select discordant pairs
discordant <- which(Y1!=Y2)
id <- rep(1:n, 2)
# Sample from discordant pairs
incl <- sample(x = discordant, size = n, replace = TRUE)
data <- data.frame(id = id, Y = c(Y1[incl], Y2[incl]), X = c(X1[incl], X2[incl]),
                   Z = c(Z1[incl], Z2[incl]))

# Estimation of the attributable fraction
AF.cc_match <- AF.cc(formula = Y ~ X + Z + X * Z, data = data,
                          exposure = "X", clusterid = "id", matched = TRUE)
summary(AF.cc_match)
```

`AF.ch` *Attributable fraction function for cohort sampling designs with time-to-event outcomes. NOTE! Deprecated function. Use* `AFcoxph`.

Description

`AF.ch` estimates the model-based adjusted attributable fraction function for data from cohort sampling designs with time-to-event outcomes.

Usage

```
AF.ch(formula, data, exposure, ties = "breslow", times, clusterid)
```

Arguments

`formula`	a formula object, with the response on the left of a ~ operator, and the terms on the right. The response must be a survival object as returned by the `Surv` function (`Surv`). The exposure and confounders should be specified as independent (right-hand side) variables. The time-to-event outcome should be specified by the survival object. The formula is used to fit a Cox proportional hazards model.
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment (`formula`), typically the environment from which the function is called.
`exposure`	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
`ties`	a character string specifying the method for tie handling. If there are no tied death times all the methods are equivalent. Uses the Breslow method by default.
`times`	a scalar or vector of time points specified by the user for which the attributable fraction function is estimated. If not specified the observed death times will be used.
`clusterid`	the name of the cluster identifier variable as a string, if data are clustered.

Details

`Af.ch` estimates the attributable fraction for a time-to-event outcome under the hypothetical scenario where a binary exposure `X` is eliminated from the population. The estimate is adjusted for confounders `Z` by the Cox proportional hazards model (`coxph`). Let the AF function be defined as

$$AF = 1 - \frac{\{1 - S_0(t)\}}{\{1 - S(t)\}}$$

where $S_0(t)$ denotes the counterfactual survival function for the event if the exposure would have been eliminated from the population at baseline and $S(t)$ denotes the factual survival function. If `Z` is sufficient for confounding control, then $S_0(t)$ can be expressed as $E_Z\{S(t \mid X = 0, Z)\}$. The function uses Cox proportional hazards regression to estimate $S(t \mid X = 0, Z)$, and the marginal sample distribution of `Z` to approximate the outer expectation (Sjölander and Vansteelandt, 2014). If `clusterid` is supplied, then a clustered sandwich formula is used in all variance calculations.

Value

`AF.est`	estimated attributable fraction function for every time point specified by `times`.
`AF.var`	estimated variance of `AF.est`. The variance is obtained by combining the delta methods with the sandwich formula.
`S.est`	estimated factual survival function; $S(t)$.
`S.var`	estimated variance of `S.est`. The variance is obtained by the sandwich formula.
`S0.est`	estimated counterfactual survival function if exposure would be eliminated; $S_0(t)$.
`S0.var`	estimated variance of `S0.est`. The variance is obtained by the sandwich formula.
`object`	the fitted model. Fitted using Cox proportional hazard, `coxph`.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Chen, L., Lin, D. Y., and Zeng, D. (2010). Attributable fraction functions for censored event times. *Biometrika* **97**, 713-726.

Sjölander, A. and Vansteelandt, S. (2014). Doubly robust estimation of attributable fractions in survival analysis. *Statistical Methods in Medical Research*. doi: 10.1177/0962280214564003.

See Also

The new and more general version of the function: `AFcoxph`. `coxph` and `Surv` used for fitting the Cox proportional hazards model.

Examples

```
# Simulate a sample from a cohort sampling design with time-to-event outcome
expit <- function(x) 1 / (1 + exp( - x))
n <- 500
time <- c(seq(from = 0.2, to = 1, by = 0.2))
Z <- rnorm(n = n)
X <- rbinom(n = n, size = 1, prob = expit(Z))
Tim <- rexp(n = n, rate = exp(X + Z))
C <- rexp(n = n, rate = exp(X + Z))
Tobs <- pmin(Tim, C)
D <- as.numeric(Tobs < C)
#Ties created by rounding
Tobs <- round(Tobs, digits = 2)

# Example 1: non clustered data from a cohort sampling design with time-to-event outcomes
data <- data.frame(Tobs, D, X,  Z)

# Estimation of the attributable fraction
AF.ch_est <- AF.ch(formula = Surv(Tobs, D) ~ X + Z + X * Z, data = data,
                   exposure = "X", times = time)
summary(AF.ch_est)

# Example 2: clustered data from a cohort sampling design with time-to-event outcomes
# Duplicate observations in order to create clustered data
id <- rep(1:n, 2)
data <- data.frame(Tobs = c(Tobs, Tobs), D = c(D, D), X = c(X, X), Z = c(Z, Z), id = id)

# Estimation of the attributable fraction
AF.ch_clust <- AF.ch(formula = Surv(Tobs, D) ~ X + Z + X * Z, data = data,
                         exposure = "X", times = time, clusterid = "id")
summary(AF.ch_clust)
plot(AF.ch_clust, CI = TRUE)
```

`AF.cs` *Attributable fraction for cross-sectional sampling designs. NOTE! Deprecated function. Use* `AFglm`.

Description

`AF.cs` estimates the model-based adjusted attributable fraction for data from cross-sectional sampling designs.

Usage

```
AF.cs(formula, data, exposure, clusterid)
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model used for adjusting for confounders. The exposure and confounders should be specified as independent (right-hand side) variables. The outcome should be specified as dependent (left-hand side) variable. The formula is used to object a logistic regression by glm.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment (formula), typically the environment from which the function is called.
exposure	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
clusterid	the name of the cluster identifier variable as a string, if data are clustered.

Details

Af.cs estimates the attributable fraction for a binary outcome Y under the hypothetical scenario where a binary exposure X is eliminated from the population. The estimate is adjusted for confounders Z by logistic regression (glm). Let the AF be defined as

$$AF = 1 - \frac{Pr(Y_0 = 1)}{Pr(Y = 1)}$$

where $Pr(Y_0 = 1)$ denotes the counterfactual probability of the outcome if the exposure would have been eliminated from the population and $Pr(Y = 1)$ denotes the factual probability of the outcome. If Z is sufficient for confounding control, then $Pr(Y_0 = 1)$ can be expressed as $E_Z\{Pr(Y = 1 \mid X = 0, Z)\}$. The function uses logistic regression to estimate $Pr(Y = 1 \mid X = 0, Z)$, and the marginal sample distribution of Z to approximate the outer expectation (Sjölander and Vansteelandt, 2012). If clusterid is supplied, then a clustered sandwich formula is used in all variance calculations.

Value

AF.est	estimated attributable fraction.
AF.var	estimated variance of AF.est. The variance is obtained by combining the delta method with the sandwich formula.
P.est	estimated factual proportion of cases; $Pr(Y = 1)$.
P.var	estimated variance of P.est. The variance is obtained by the sandwich formula.
P0.est	estimated counterfactual proportion of cases if exposure would be eliminated; $Pr(Y_0 = 1)$.
P0.var	estimated variance of P0.est. The variance is obtained by the sandwich formula.
object	the fitted model. Fitted using logistic regression, glm.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Greenland, S. and Drescher, K. (1993). Maximum Likelihood Estimation of the Attributable Fraction from logistic Models. *Biometrics* **49**, 865-872.

Sjölander, A. and Vansteelandt, S. (2011). Doubly robust estimation of attributable fractions. *Biostatistics* **12**, 112-121.

See Also

The new and more general version of the function: `AFglm`.

Examples

```
# Simulate a cross-sectional sample
expit <- function(x) 1 / (1 + exp( - x))
n <- 1000
Z <- rnorm(n = n)
X <- rbinom(n = n, size = 1, prob = expit(Z))
Y <- rbinom(n = n, size = 1, prob = expit(Z + X))

# Example 1: non clustered data from a cross-sectional sampling design
data <- data.frame(Y, X, Z)

# Estimation of the attributable fraction
AF.cs_est <- AF.cs(formula = Y ~ X + Z + X * Z, data = data, exposure = "X")
summary(AF.cs_est)

# Example 2: clustered data from a cross-sectional sampling design
# Duplicate observations in order to create clustered data
id <- rep(1:n, 2)
data <- data.frame(id = id, Y = c(Y, Y), X = c(X, X), Z = c(Z, Z))

# Estimation of the attributable fraction
AF.cs_clust <- AF.cs(formula = Y ~ X + Z + X * Z, data = data,
                          exposure = "X", clusterid = "id")
summary(AF.cs_clust)
```

`AFclogit` *Attributable fraction estimation based on a conditional logistic regression model as a* `clogit` *object (commonly used for matched case-control sampling designs).*

Description

`AFclogit` estimates the model-based adjusted attributable fraction from a conditional logistic regression model in form of a `clogit` object. This model is model is commonly used for data from matched case-control sampling designs.

Usage

```
AFclogit(object, data, exposure, clusterid)
```

Arguments

`object`	a fitted conditional logistic regression model object of class "`clogit`".
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in `data`, the variables are taken from environment (`formula`), typically the environment from which the function is called.
`exposure`	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
`clusterid`	the name of the cluster identifier variable as a string. Because conditional logistic regression is only used for clustered data, this argument must be supplied.

Details

`AFclogit` estimates the attributable fraction for a binary outcome `Y` under the hypothetical scenario where a binary exposure `X` is eliminated from the population. The estimate is adjusted for confounders `Z` by conditional logistic regression. The estimation assumes that the outcome is rare so that the risk ratio can be approximated by the odds ratio, for details see Bruzzi et. al. Let the AF be defined as

$$AF = 1 - \frac{Pr(Y_0 = 1)}{Pr(Y = 1)}$$

where $Pr(Y_0 = 1)$ denotes the counterfactual probability of the outcome if the exposure would have been eliminated from the population. If `Z` is sufficient for confounding control then the probability $Pr(Y_0 = 1)$ can be expressed as

$$Pr(Y_0 = 1) = E_Z\{Pr(Y = 1 \mid X = 0, Z)\}.$$

Using Bayes' theorem this implies that the AF can be expressed as

$$AF = 1 - \frac{E_Z\{Pr(Y = 1 \mid X = 0, Z)\}}{Pr(Y = 1)} = 1 - E_Z\{RR^{-X}(Z) \mid Y = 1\}$$

where $RR(Z)$ is the risk ratio

$$\frac{Pr(Y = 1 \mid X = 1, Z)}{Pr(Y = 1 \mid X = 0, Z)}.$$

Moreover, the risk ratio can be approximated by the odds ratio if the outcome is rare. Thus,

$$AF \approx 1 - E_Z\{OR^{-X}(Z) \mid Y = 1\}.$$

The odds ratio is estimated by conditional logistic regression. The function `gee` in the `drgee` package is used to get the score contributions for each cluster and the hessian. A clustered sandwich formula is used in the variance calculation.

Value

`AF.est` estimated attributable fraction.

`AF.var` estimated variance of `AF.est`. The variance is obtained by combining the delta methods with the sandwich formula.

`log.or` a vector of the estimated log odds ratio for every individual. `log.or` contains the estimated coefficient for the exposure variable `X` for every level of the confounder `Z` as specified by the user in the formula. If the model to be estimated is

$$logit\{Pr(Y = 1|X, Z)\} = \alpha + \beta X + \gamma Z$$

then `log.or` is the estimate of β. If the model to be estimated is

$$logit\{Pr(Y = 1|X, Z)\} = \alpha + \beta X + \gamma Z + \psi XZ$$

then `log.odds` is the estimate of $\beta + \psi Z$.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Bruzzi, P., Green, S. B., Byar, D., Brinton, L. A., and Schairer, C. (1985). Estimating the population attributable risk for multiple risk factors using case-control data. *American Journal of Epidemiology* **122**, 904-914.

See Also

`clogit` used for fitting the conditional logistic regression model for matched case-control designs. For non-matched case-control designs see `AFglm`.

Examples

```
expit <- function(x) 1 / (1 + exp( - x))
NN <- 1000000
n <- 500

# Example 1: matched case-control
# Duplicate observations in order to create a matched data sample
# Create an unobserved confounder U common for each pair of individuals
intercept <- -6
U  <- rnorm(n = NN)
Z1 <- rnorm(n = NN)
Z2 <- rnorm(n = NN)
X1 <- rbinom(n = NN, size = 1, prob = expit(U + Z1))
X2 <- rbinom(n = NN, size = 1, prob = expit(U + Z2))
Y1 <- rbinom(n = NN, size = 1, prob = expit(intercept + U + Z1 + X1))
Y2 <- rbinom(n = NN, size = 1, prob = expit(intercept + U + Z2 + X2))
# Select discordant pairs
discordant <- which(Y1!=Y2)
id <- rep(1:n, 2)
```

```
# Sample from discordant pairs
incl <- sample(x = discordant, size = n, replace = TRUE)
data <- data.frame(id = id, Y = c(Y1[incl], Y2[incl]), X = c(X1[incl], X2[incl]),
                   Z = c(Z1[incl], Z2[incl]))

# Fit a clogit object
fit <- clogit(formula = Y ~ X + Z + X * Z + strata(id), data = data)

# Estimate the attributable fraction from the fitted conditional logistic regression
AFclogit_est <- AFclogit(fit, data, exposure = "X", clusterid="id")
summary(AFclogit_est)
```

AFcoxph *Attributable fraction function based on a Cox Proportional Hazard regression model as a* coxph *object (commonly used for cohort sampling designs with time-to-event outcomes).*

Description

AFcoxph estimates the model-based adjusted attributable fraction function from a Cox Proportional Hazard regression model in form of a coxph object. This model is commonly used for data from cohort sampling designs with time-to-event outcomes.

Usage

```
AFcoxph(object, data, exposure, times, clusterid)
```

Arguments

object	a fitted Cox Proportional Hazard regression model object of class "coxph". Method for handling ties must be breslow since this is assumed in the calculation of the standard errors. No special terms such as cluster, strata and tt is allowed in the formula for the fitted object.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment (formula), typically the environment from which the function is called.
exposure	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
times	a scalar or vector of time points specified by the user for which the attributable fraction function is estimated. If not specified the observed event times will be used.
clusterid	the name of the cluster identifier variable as a string, if data are clustered. Cluster robust standard errors will be calculated.

Details

AFcoxph estimates the attributable fraction for a time-to-event outcome under the hypothetical scenario where a binary exposure X is eliminated from the population. The estimate is adjusted for confounders Z by the Cox proportional hazards model (coxph). Let the AF function be defined as

$$AF = 1 - \frac{\{1 - S_0(t)\}}{\{1 - S(t)\}}$$

where $S_0(t)$ denotes the counterfactual survival function for the event if the exposure would have been eliminated from the population at baseline and $S(t)$ denotes the factual survival function. If Z is sufficient for confounding control, then $S_0(t)$ can be expressed as $E_Z\{S(t \mid X = 0, Z)\}$. The function uses a fitted Cox proportional hazards regression to estimate $S(t \mid X = 0, Z)$, and the marginal sample distribution of Z to approximate the outer expectation (Sjölander and Vansteelandt, 2014). If clusterid is supplied, then a clustered sandwich formula is used in all variance calculations.

Value

AF.est	estimated attributable fraction function for every time point specified by times.
AF.var	estimated variance of AF.est. The variance is obtained by combining the delta methods with the sandwich formula.
S.est	estimated factual survival function; $S(t)$.
S.var	estimated variance of S.est. The variance is obtained by the sandwich formula.
S0.est	estimated counterfactual survival function if exposure would be eliminated; $S_0(t)$.
S0.var	estimated variance of S0.est. The variance is obtained by the sandwich formula.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Chen, L., Lin, D. Y., and Zeng, D. (2010). Attributable fraction functions for censored event times. *Biometrika* **97**, 713-726.

Sjölander, A. and Vansteelandt, S. (2014). Doubly robust estimation of attributable fractions in survival analysis. *Statistical Methods in Medical Research*. doi: 10.1177/0962280214564003.

See Also

coxph and Surv used for fitting the Cox proportional hazards model.

Examples

```
# Simulate a sample from a cohort sampling design with time-to-event outcome
expit <- function(x) 1 / (1 + exp( - x))
n <- 500
time <- c(seq(from = 0.2, to = 1, by = 0.2))
```

```
Z <- rnorm(n = n)
X <- rbinom(n = n, size = 1, prob = expit(Z))
Tim <- rexp(n = n, rate = exp(X + Z))
C <- rexp(n = n, rate = exp(X + Z))
Tobs <- pmin(Tim, C)
D <- as.numeric(Tobs < C)
#Ties created by rounding
Tobs <- round(Tobs, digits = 2)

# Example 1: non clustered data from a cohort sampling design with time-to-event outcomes
data <- data.frame(Tobs, D, X,  Z)

# Fit a Cox PH regression model
fit <- coxph(formula = Surv(Tobs, D) ~ X + Z + X * Z, data = data, ties="breslow")

# Estimate the attributable fraction from the fitted Cox PH regression model
AFcoxph_est <- AFcoxph(fit, data=data, exposure ="X", times = time)
summary(AFcoxph_est)

# Example 2: clustered data from a cohort sampling design with time-to-event outcomes
# Duplicate observations in order to create clustered data
id <- rep(1:n, 2)
data <- data.frame(Tobs = c(Tobs, Tobs), D = c(D, D), X = c(X, X), Z = c(Z, Z), id = id)

# Fit a Cox PH regression model
fit <- coxph(formula = Surv(Tobs, D) ~ X + Z + X * Z, data = data, ties="breslow")

# Estimate the attributable fraction from the fitted Cox PH regression model
AFcoxph_clust <- AFcoxph(object = fit, data = data,
                         exposure = "X", times = time, clusterid = "id")
summary(AFcoxph_clust)
plot(AFcoxph_clust, CI = TRUE)

# Estimate the attributable fraction from the fitted Cox PH regression model, time unspecified
AFcoxph_clust_no_time <- AFcoxph(object = fit, data = data,
                         exposure = "X", clusterid = "id")
summary(AFcoxph_clust_no_time)
plot(AFcoxph_clust, CI = TRUE)
```

AFglm *Attributable fraction estimation based on a logistic regression model from a* glm *object (commonly used for cross-sectional or case-control sampling designs).*

Description

AFglm estimates the model-based adjusted attributable fraction for data from a logistic regression model in the form of a glm object. This model is commonly used for data from a cross-sectional or non-matched case-control sampling design.

Usage

```
AFglm(object, data, exposure, clusterid, case.control = FALSE)
```

Arguments

`object`	a fitted logistic regression model object of class "glm".
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment (`formula`), typically the environment from which the function is called.
`exposure`	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
`clusterid`	the name of the cluster identifier variable as a string, if data are clustered. Cluster robust standard errors will be calculated.
`case.control`	can be set to `TRUE` if the data is from a non-matched case control study. By default `case.control` is set to `FALSE` which is used for cross-sectional sampling designs.

Details

`AFglm` estimates the attributable fraction for a binary outcome `Y` under the hypothetical scenario where a binary exposure `X` is eliminated from the population. The estimate is adjusted for confounders `Z` by logistic regression using the (`glm`) function. The estimation strategy is different for cross-sectional and case-control sampling designs even if the underlying logististic regression model is the same. For cross-sectional sampling designs the AF can be defined as

$$AF = 1 - \frac{Pr(Y_0 = 1)}{Pr(Y = 1)}$$

where $Pr(Y_0 = 1)$ denotes the counterfactual probability of the outcome if the exposure would have been eliminated from the population and $Pr(Y = 1)$ denotes the factual probability of the outcome. If `Z` is sufficient for confounding control, then $Pr(Y_0 = 1)$ can be expressed as $E_Z\{Pr(Y = 1 \mid X = 0, Z)\}$. The function uses logistic regression to estimate $Pr(Y = 1 \mid X = 0, Z)$, and the marginal sample distribution of `Z` to approximate the outer expectation (Sjölander and Vansteelandt, 2012). For case-control sampling designs the outcome prevalence is fixed by sampling design and absolute probabilities (`P.est` and `P0.est`) can not be estimated. Instead adjusted log odds ratios (`log.or`) are estimated for each individual. This is done by setting `case.control` to `TRUE`. It is then assumed that the outcome is rare so that the risk ratio can be approximated by the odds ratio. For case-control sampling designs the AF be defined as (Bruzzi et. al)

$$AF = 1 - \frac{Pr(Y_0 = 1)}{Pr(Y = 1)}$$

where $Pr(Y_0 = 1)$ denotes the counterfactual probability of the outcome if the exposure would have been eliminated from the population. If `Z` is sufficient for confounding control then the probability $Pr(Y_0 = 1)$ can be expressed as

$$Pr(Y_0 = 1) = E_Z\{Pr(Y = 1 \mid X = 0, Z)\}.$$

Using Bayes' theorem this implies that the AF can be expressed as

$$AF = 1 - \frac{E_Z\{Pr(Y = 1 \mid X = 0, Z)\}}{Pr(Y = 1)} = 1 - E_Z\{RR^{-X}(Z) \mid Y = 1\}$$

where $RR(Z)$ is the risk ratio

$$\frac{Pr(Y = 1 \mid X = 1, Z)}{Pr(Y = 1 \mid X = 0, Z)}.$$

Moreover, the risk ratio can be approximated by the odds ratio if the outcome is rare. Thus,

$$AF \approx 1 - E_Z\{OR^{-X}(Z) \mid Y = 1\}.$$

If `clusterid` is supplied, then a clustered sandwich formula is used in all variance calculations.

Value

`AF.est` estimated attributable fraction.

`AF.var` estimated variance of `AF.est`. The variance is obtained by combining the delta method with the sandwich formula.

`P.est` estimated factual proportion of cases; $Pr(Y = 1)$. Returned by default when `case.control = FALSE`.

`P.var` estimated variance of `P.est`. The variance is obtained by the sandwich formula. Returned by default when `case.control = FALSE`.

`P0.est` estimated counterfactual proportion of cases if exposure would be eliminated; $Pr(Y_0 = 1)$. Returned by default when `case.control = FALSE`.

`P0.var` estimated variance of `P0.est`. The variance is obtained by the sandwich formula. Returned by default when `case.control = FALSE`.

`log.or` a vector of the estimated log odds ratio for every individual. `log.or` contains the estimated coefficient for the exposure variable `X` for every level of the confounder `Z` as specified by the user in the formula. If the model to be estimated is

$$logit\{Pr(Y = 1|X, Z)\} = \alpha + \beta X + \gamma Z$$

then `log.or` is the estimate of β. If the model to be estimated is

$$logit\{Pr(Y = 1|X, Z)\} = \alpha + \beta X + \gamma Z + \psi XZ$$

then `log.odds` is the estimate of $\beta+\psi Z$. Only returned if argument `case.control` is set to `TRUE`.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Bruzzi, P., Green, S. B., Byar, D., Brinton, L. A., and Schairer, C. (1985). Estimating the population attributable risk for multiple risk factors using case-control data. *American Journal of Epidemiology* **122**, 904-914.

Greenland, S. and Drescher, K. (1993). Maximum Likelihood Estimation of the Attributable Fraction from logistic Models. *Biometrics* **49**, 865-872.

Sjölander, A. and Vansteelandt, S. (2011). Doubly robust estimation of attributable fractions. *Biostatistics* **12**, 112-121.

See Also

`glm` used for fitting the logistic regression model. For conditional logistic regression (commonly for data from a matched case-control sampling design) see `AFclogit`.

Examples

```
# Simulate a cross-sectional sample

expit <- function(x) 1 / (1 + exp( - x))
n <- 1000
Z <- rnorm(n = n)
X <- rbinom(n = n, size = 1, prob = expit(Z))
Y <- rbinom(n = n, size = 1, prob = expit(Z + X))

# Example 1: non clustered data from a cross-sectional sampling design
data <- data.frame(Y, X, Z)

# Fit a glm object
fit <- glm(formula = Y ~ X + Z + X * Z, family = binomial, data = data)

# Estimate the attributable fraction from the fitted logistic regression
AFglm_est <- AFglm(object = fit, data = data, exposure = "X")
summary(AFglm_est)

# Example 2: clustered data from a cross-sectional sampling design
# Duplicate observations in order to create clustered data
id <- rep(1:n, 2)
data <- data.frame(id = id, Y = c(Y, Y), X = c(X, X), Z = c(Z, Z))

# Fit a glm object
fit <- glm(formula = Y ~ X + Z + X * Z, family = binomial, data = data)

# Estimate the attributable fraction from the fitted logistic regression
AFglm_clust <- AFglm(object = fit, data = data,
                         exposure = "X", clusterid = "id")
summary(AFglm_clust)


# Example 3: non matched case-control
# Simulate a sample from a non matched case-control sampling design
# Make the outcome a rare event by setting the intercept to -6
```

```
expit <- function(x) 1 / (1 + exp( - x))
NN <- 1000000
n <- 500
intercept <- -6
Z <- rnorm(n = NN)
X <- rbinom(n = NN, size = 1, prob = expit(Z))
Y <- rbinom(n = NN, size = 1, prob = expit(intercept + X + Z))
population <- data.frame(Z, X, Y)
Case <- which(population$Y == 1)
Control <- which(population$Y == 0)
# Sample cases and controls from the population
case <- sample(Case, n)
control <- sample(Control, n)
data <- population[c(case, control), ]

# Fit a glm object
fit <- glm(formula = Y ~ X + Z + X * Z, family = binomial, data = data)

# Estimate the attributable fraction from the fitted logistic regression
AFglm_est_cc <- AFglm(object = fit, data = data, exposure = "X", case.control = TRUE)
summary(AFglm_est_cc)
```

`AFivglm` *Attributable fraction function based on Instrumental Variables (IV) regression as an* `ivglm` *object in the* `ivtools` *package.*

Description

`AFivglm` estimates the model-based adjusted attributable fraction from a Instrumental Variable regression from a `ivglm` object. The IV regression can be estimated by either G-estimation or Two Stage estimation for a binary exposure and outcome.

Usage

```
AFivglm(object, data)
```

Arguments

`object`	a fitted Instrumental Variable regression of class "`ivglm`".
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment (`formula`), typically the environment from which the function is called.

Details

AFivglm estimates the attributable fraction for an IV regression under the hypothetical scenario where a binary exposure X is eliminated from the population. The estimate can be adjusted for IV-outcome confounders L in the ivglm function. Let the AF function be defined as

$$AF = 1 - \frac{Pr(Y_0 = 1)}{Pr(Y = 1)}$$

where $Pr(Y_0 = 1)$ denotes the counterfactual outcome prevalence had everyone been unexposed and $Pr(Y = 1)$ denotes the factual outcome prevalence. If the instrument Z is valid, conditional on covariates L, i.e. fulfills the IV assumptions 1) the IV should have a (preferably strong) association with the exposure, 2) the effect of the IV on the outcome should only go through the exposure and 3) the IV-outcome association should be unconfounded (Imbens and Angrist, 1994) then $Pr(Y_0 = 1)$ can be estimated.

Value

AF.est	estimated attributable fraction.
AF.var	estimated variance of AF.est. The variance is obtained by combining the delta methods with the sandwich formula.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

References

Dahlqwist E., Kutalik Z., Sjölander, A. (2019). Using Instrumental Variables to estimate the attributable fraction. *Manuscript*.

See Also

ivglm used for fitting the causal risk ratio or odds ratio using the G-estimator or Two stage estimator.

Examples

```
# Example 1
set.seed(2)
n <- 5000
## parameter a0 determines the outcome prevalence
a0 <- -4
psi.true <- 1
l <- rbinom(n, 1, 0.5)
u <- rbinom(n, 1, 0.5)
z <- rbinom(n, 1, plogis(a0))
x <- rbinom(n, 1, plogis(a0+3*z+ u))
y <- rbinom(n, 1, exp(a0+psi.true*x+u))
d <- data.frame(z,u,x,y,l)
## Outcome prevalence
mean(d$y)
```

```
####### G-estimation
## log CRR
fitz.l <- glm(z~1, family=binomial, data=d)
gest_log <- ivglm(estmethod="g", X="x", Y="y",
                  fitZ.L=fitz.l, data=d, link="log")
AFgestlog <- AFivglm(gest_log, data=d)
summary(AFgestlog)

## log COR
## Associational model, saturated
fit_y <- glm(y~x+z+x*z, family="binomial", data=d)
## Estimations of COR and AF
gest_logit <- ivglm(estmethod="g", X="x", Y="y",
                    fitZ.L=fitz.l, fitY.LZX=fit_y,
                    data=d, link="logit")
AFgestlogit <- AFivglm(gest_logit, data = d)
summary(AFgestlogit)

####### TS estimation
## log CRR
# First stage
fitx <- glm(x ~ z, family=binomial, data=d)
# Second stage
fity <- glm(y ~ x, family=poisson, data=d)
## Estimations of CRR and AF
TSlog <- ivglm(estmethod="ts", X="x", Y="y",
               fitY.LX=fity, fitX.LZ=fitx, data=d, link="log")
AFtslog <- AFivglm(TSlog, data=d)
summary(AFtslog)

## log COR
# First stage
fitx_logit <- glm(x ~ z, family=binomial, data=d)
# Second stage
fity_logit <- glm(y ~ x, family=binomial, data=d)
## Estimations of COR and AF
TSlogit <- ivglm(estmethod="ts", X="x", Y="y",
                 fitY.LX=fity_logit, fitX.LZ=fitx_logit,
                  data=d, link="logit")
AFtslogit <- AFivglm(TSlogit, data=d)
summary(AFtslogit)

## Example 2: IV-outcome confounding by L
####### G-estimation
## log CRR
fitz.l <- glm(z~l, family=binomial, data=d)
gest_log <- ivglm(estmethod="g", X="x", Y="y",
                  fitZ.L=fitz.l, data=d, link="log")
AFgestlog <- AFivglm(gest_log, data=d)
summary(AFgestlog)

## log COR
```

```
## Associational model
fit_y <- glm(y~x+z+l+x*z+x*l+z*l, family="binomial", data=d)
## Estimations of COR and AF
gest_logit <- ivglm(estmethod="g", X="x", Y="y",
                    fitZ.L=fitz.l, fitY.LZX=fit_y,
                    data=d, link="logit")
AFgestlogit <- AFivglm(gest_logit, data = d)
summary(AFgestlogit)

####### TS estimation
## log CRR
# First stage
fitx <- glm(x ~ z+l, family=binomial, data=d)
# Second stage
fity <- glm(y ~ x+l, family=poisson, data=d)
## Estimations of CRR and AF
TSlog <- ivglm(estmethod="ts", X="x", Y="y",
               fitY.LX=fity, fitX.LZ=fitx, data=d,
               link="log")
AFtslog <- AFivglm(TSlog, data=d)
summary(AFtslog)

## log COR
# First stage
fitx_logit <- glm(x ~ z+l, family=binomial, data=d)
# Second stage
fity_logit <- glm(y ~ x+l, family=binomial, data=d)
## Estimations of COR and AF
TSlogit <- ivglm(estmethod="ts", X="x", Y="y",
                 fitY.LX=fity_logit, fitX.LZ=fitx_logit,
                 data=d, link="logit")
AFtslogit <- AFivglm(TSlogit, data=d)
summary(AFtslogit)
```

`AFparfrailty` *Attributable fraction function based on a Weibull gamma-frailty model as a* `parfrailty` *object (commonly used for cohort sampling family designs with time-to-event outcomes).*

Description

`AFparfrailty` estimates the model-based adjusted attributable fraction function from a shared Weibull gamma-frailty model in form of a `parfrailty` object. This model is commonly used for data from cohort sampling familty designs with time-to-event outcomes.

Usage

```
AFparfrailty(object, data, exposure, times, clusterid)
```

Arguments

object	a fitted Weibull gamma-parfrailty object of class "parfrailty".
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment (formula), typically the environment from which the function is called.
exposure	the name of the exposure variable as a string. The exposure must be binary (0/1) where unexposed is coded as 0.
times	a scalar or vector of time points specified by the user for which the attributable fraction function is estimated. If not specified the observed death times will be used.
clusterid	the name of the cluster identifier variable as a string, if data are clustered.

Details

AFparfrailty estimates the attributable fraction for a time-to-event outcome under the hypothetical scenario where a binary exposure X is eliminated from the population. The estimate is adjusted for confounders Z by the shared frailty model (parfrailty). The baseline hazard is assumed to follow a Weibull distribution and the unobserved shared frailty effects U are assumed to be gamma distributed. Let the AF function be defined as

$$AF = 1 - \frac{\{1 - S_0(t)\}}{\{1 - S(t)\}}$$

where $S_0(t)$ denotes the counterfactual survival function for the event if the exposure would have been eliminated from the population at baseline and $S(t)$ denotes the factual survival function. If Z and U are sufficient for confounding control, then $S_0(t)$ can be expressed as $E_Z\{S(t \mid X = 0, Z)\}$. The function uses a fitted Weibull gamma-frailty model to estimate $S(t \mid X = 0, Z)$, and the marginal sample distribution of Z to approximate the outer expectation. A clustered sandwich formula is used in all variance calculations.

Value

AF.est	estimated attributable fraction function for every time point specified by times.
AF.var	estimated variance of AF.est. The variance is obtained by combining the delta methods with the sandwich formula.
S.est	estimated factual survival function; $S(t)$.
S.var	estimated variance of S.est. The variance is obtained by the sandwich formula.
S0.est	estimated counterfactual survival function if exposure would be eliminated; $S_0(t)$.
S0.var	estimated variance of S0.est. The variance is obtained by the sandwich formula.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

See Also

parfrailty used for fitting the Weibull gamma-frailty and stdParfrailty used for standardization of a parfrailty object.

Examples

```
# Example 1: clustered data with frailty U
expit <- function(x) 1 / (1 + exp( - x))
n <- 100
m <- 2
alpha <- 1.5
eta <- 1
phi <- 0.5
beta <- 1
id <- rep(1:n,each=m)
U <- rep(rgamma(n, shape = 1 / phi, scale = phi), each = m)
Z <- rnorm(n * m)
X <- rbinom(n * m, size = 1, prob = expit(Z))
# Reparametrize scale as in rweibull function
weibull.scale <- alpha / (U * exp(beta * X)) ^ (1 / eta)
t <- rweibull(n * m, shape = eta, scale = weibull.scale)

# Right censoring
c <- runif(n * m, 0, 10)
delta <- as.numeric(t < c)
t <- pmin(t, c)

data <- data.frame(t, delta, X, Z, id)

# Fit a parfrailty object
library(stdReg)
fit <- parfrailty(formula = Surv(t, delta) ~ X + Z + X * Z, data = data, clusterid = "id")
summary(fit)

# Estimate the attributable fraction from the fitted frailty model

time <- c(seq(from = 0.2, to = 1, by = 0.2))

AFparfrailty_est <- AFparfrailty(object = fit, data = data, exposure = "X",
                                 times = time, clusterid = "id")
summary(AFparfrailty_est)
plot(AFparfrailty_est, CI = TRUE, ylim=c(0.1,0.7))
```

---

clslowbwt *Birthweight data clustered on the mother.*

Description

This dataset is borrowed from "An introduction to Stata for health reserachers" (Juul and Frydenberg, 2010). The dataset contains data on 189 mothers who have given birth to one or several children. In total, the dataset contains data on 487 births.

Usage

```
data(clslowbwt)
```

Format

The dataset is structured so that each row corresponds to one birth/child. It contains the following variables:

id the identification number of the mother.

birth the number of the birth, i.e. "1" for the mother's first birth, "2" for the mother's second birth etc.

smoke a categorical variable indicating if the mother is a smoker or not with levels "0. No" and "1. Yes".

race the race of the mother with levels "1. White", "2. Black" or "3. Other".

age the age of the mother at childbirth.

lwt weight of the mother at last menstruational period (in pounds).

bwt birthweight of the newborn.

low a categorical variable indicating if the newborn is categorized as a low birthweight baby (<2500 grams) or not with levels "0. No" and "1. Yes".

smoker a numeric indicator if the mother is a smoker or not. Recoded version of the variable "smoke" where "0.No" is recoded as "0" and "1.Yes" is recoded as "1".

lbw a numeric indicator of whether the newborn is categorized as a low birthweight baby (<2500 grams) or not. Recoded version of the variable "low" where "0.No" is recoded as "0" and "1.Yes" is recoded as "1".

The following changes have been made to the original data in Juul & Frydenberg (2010):

- The variable "low" is recoded into the numeric indicator variable "lbw":

```
clslowbwt$lbw <- as.numeric(clslowbwt$low == "1. Yes")
```

- The variable "smoke" is recoded into the numeric indicator variable "smoker":

```
clslowbwt$smoker <- as.numeric(clslowbwt$smoke == "1. Yes")
```

References

Juul, Svend & Frydenberg, Morten (2010). *An introduction to Stata for health researchers*, Texas, Stata press, 2010 (Third edition).

http://www.stata-press.com/data/ishr3.html

plot.AF — *Plot function for objects of class "AF" from the function* AFcoxph *or* AFparfrailty.

Description

Creates a simple scatterplot for the AF function with time sequence (specified by the user as times in the AFcoxph function) on the x-axis and the AF function estimate on the y-axis.

Usage

```
## S3 method for class 'AF'
plot(x, CI = TRUE, confidence.level, CI.transform, xlab,
  main, ylim, ...)
```

Arguments

x	an object of class AF from the AFcoxph or AFparfrailty function.
CI	if TRUE confidence intervals are estimated and ploted in the graph.
confidence.level	user-specified confidence level for the confidence intervals. If not specified it defaults to 95 percent. Should be specified in decimals such as 0.95 for 95 percent.
CI.transform	user-specified transformation of the Wald confidence interval(s). Options are untransformed, log and logit. If not specified untransformed will be calculated.
xlab	label on the x-axis. If not specified the label *"Time"* will be displayed.
main	main title of the plot. If not specified the lable *"Estimate of the attributable fraction function"* will be displayed.
ylim	limits on the y-axis of the plot. If not specified the minimum value of the lower bound of the confidence interval will be used as the minimal value and the maximum value of the upper bound of the confidence interval will be used as the maximum of y-axis of the plot.
...	further arguments to be passed to the plot function. See plot.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

rott2 *Cohort study on breast cancer patients from the Netherlands.*

Description

This dataset is borrowed from "Flexible parametric survival analysis using Stata: beyond the Cox model" (Roystone and Lambert, 2011). It contains follow-up data on 2982 woman with breast cancer who have gone through breast surgery. The women are followed from the time of surgery until death, relapse or censoring.

Usage

```
data(rott2)
```

Format

The dataset `rott2` contains the following variables:

pid patient ID number.

year year of breast surgery (i.e. year of enrollment into the study), between the years 1978-1993.

rf relapse free interval measured in months.

rfi relapse indicator.

m metastasis free.

mfi metastasis status.

os overall survival

osi overall survival indicator

age age at surgery measured in years.

meno menopausal status with levels "`pre`" and "`post`".

size tumor size in three classes: `<=20mm`, `>20-50mmm` and `>50mm`.

grade differentiation grade with levels 2 or 3.

pr progesterone receptors, fmol/l.

er oestrogen receptors, fmol/l.

nodes the number of positive lymph nodes.

hormon hormonal therapy with levels "`no`" and "`yes`".

chemo categorical variable indicating whether the patient recieved chemotheraphy or not, with levels "`no`" and "`yes`".

recent a numeric indicator of whether the tumor was discovered recently with levels "`1978-87`" and "`1988-93`".

no.chemo a numerical indicator of whether the patient did not recieved chemotherapy. Recoded version of "`chemo`" where "`yes`" is recoded as 0 and "`no`" is recoded as 1.

The following changes have been made to the original data in Roystone and Lambert (2011):

- The variable "chemo" is recoded into the numeric indicator variable "no.chemo":

```
rott22$no.chemo <- as.numeric(rott2$chemo == "no")
```

The follwing variables have been removed from the original dataset: enodes, pr_1, enodes_1, _st, _d, _t, _t0 since they are recodings of some existing variables which are not used in this analysis.

References

Royston, Patrick & Lambert, Paul. C (2011). *Flexible parametric survival analysis using Stata: beyond the Cox model*. College Station, Texas, U.S, Stata press.

http://www.stata-press.com/data/fpsaus.html

---

singapore *Case-control study on oesophageal cancer in Chinese Singapore men.*

Description

This dataset is borrowed from "Aetiological factors in oesophageal cancer in Singapore Chinese" by De Jong UW, Breslow N, Hong JG, Sridharan M, Shanmugaratnam K (1974).

Usage

```
data(singapore)
```

Format

The dataset contains the following variables:

Age age of the patient.

Dial dialect group where 1 represent "Hokhien/Teochew" and 0 represent "Cantonese/Other".

Samsu a numeric indicator of whether the patient consumes Samsu wine or not.

Cigs number of cigarettes smoked per day.

Bev number of beverage at "burning hot" temperatures ranging between 0 to 3 different drinks per day.

Everhotbev a numeric indicator of whether the patients ever drinks "burning hot beverage" or not. Recoded from the variable "Bev".

Set matched set identification number.

CC a numeric variable where 1 represent if the patient is a case, 2 represent if the patient is a control from the same ward as the case and 3 represent if the patient is control from orthopedic hospital.

Oesophagealcancer a numeric indicator variable of whether the patient is a case of oesophageal cancer or not.

The following changes have been made to the data from the original data in De Jong UW (1974):

- The variable "Bev" is recoded into the numeric indicator variable "Everhotbev":

```
singapore$Everhotbev <- ifelse(singapore$Bev >= 1, 1, 0)
```

References

De Jong UW, Breslow N, Hong JG, Sridharan M, Shanmugaratnam K. (1974). Aetiological factors in oesophageal cancer in Singapore Chinese. *Int J Cancer* Mar 15;13(3), 291-303.

http://faculty.washington.edu/heagerty/Courses/b513/WEB2002/datasets.html

---

`summary.AF` *Summary function for objects of class "AF".*

Description

Gives a summary of the AF estimate(s) including z-value, p-value and confidence interval(s).

Usage

```
## S3 method for class 'AF'
summary(object, digits = max(3L, getOption("digits") - 3L),
  confidence.level, CI.transform, ...)
```

Arguments

`object`	an object of class `AF` from `AFglm`, `AFcoxph`, `AFclogit`, `AFparfrailty` or `AFivglm` functions.
`digits`	maximum number of digits.
`confidence.level`	user-specified confidence level for the confidence intervals. If not specified it defaults to 95 percent. Should be specified in decimals such as 0.95 for 95 percent.
`CI.transform`	user-specified transformation of the Wald confidence interval(s). Options are `untransformed`, `log` and `logit`. If not specified untransformed will be calculated.
`...`	further arguments to be passed to the summary function. See `summary`.

Author(s)

Elisabeth Dahlqwist, Arvid Sjölander

Index