Package ‘geecure’

July 22, 2025

Type Package

Title Marginal Proportional Hazards Mixture Cure Models with Generalized Estimating Equations

Version 1.0-6

Date 2018-3-28

Author Yi Niu [aut, cre],
Hui Song [ctb],
Xiaoguang Wang [ctb],
Yingwei Peng [aut, ctb]

Maintainer Yi Niu <niuyi@dlut.edu.cn>

Description Features the marginal parametric and semi-parametric proportional hazards mixture cure models for analyzing clustered survival data with a possible cure fraction. A reference is Yi Niu and Yingwei Peng (2014) <doi:10.1016/j.jmva.2013.09.003>.

License GPL (>= 2)

Imports Matrix, MASS, geepack, methods

Depends survival

LazyData TRUE

Repository CRAN

NeedsCompilation no

Date/Publication 2018-04-01 09:53:51 UTC

Contents

`geecure-package` *Marginal proportional hazards mixture cure models with generalzied estimating equations*

Description

A package that uses generalized estimating equations (GEE) approach to estimate marginal proportional hazards mixture cure (PHMC) models. This package implements recently developed inference procedures for the marginal PHMC models with the expectation-solution (ES) algorithm. The package includes the parametric PHMC model with Weibull baseline distribution in the latency part and the semiparametric PHMC model for fitting the multivariate survival data with a cure fraction.

Details

Package: geecure
Type: Package
Version: 1.0-6
Date: 2018-03-28
License: GPL(>=2)
LazyData: TRUE

Author(s)

Yi Niu <niuyi@dlut.edu.cn>, Hui Song, Xiaoguang Wang, Yingwei Peng

References

Liang, K.-Y. and Zeger, S. L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73**: 13-22.

Niu, Y. and Peng, Y. (2013) A semiparametric marginal mixture cure model for clustered survival data. *Statistics in Medicine*, **32**: 2364-2373.

Niu, Y. and Peng, Y. (2014) Marginal regression analysis of clustered failure time data with a cure fraction. *Journal of Multivariate Analysis*, **123**: 129-142.

Niu, Y., Song, L., Liu, Y, and Peng, Y. (2018) Modeling clustered long-term survivors using marginal mixture cure model. *Biometrical Journal*, doi: 10.1002/bjmj.201700114.

Peng, Y., Taylor, J. M. G, and Yu, B. (2007) A marginal regression model for multivariate failure time data with a surviving fraction. *Lifetime Data Analysis*, **13**: 351-369

Rosen, O., Jiang, W., and Tanner, M. A. (2000) Mixtures of marginal models. *Biometrika*, **87**: 391-404.

Yu, B. and Peng, Y. (2008) Mixture cure models for multivariate survival data. *Computational Statistics & Data Analysis*, **52**: 1524-1532.

`basesurv` *Estimation of the baseline survival function*

Description

The estimated baseline survival function based on the product-limit estimator (Kalbfleisch and Prentice, 2002), which is uesd to update the E-step in the ES algorithm.

Usage

```
basesurv(Time, Status, X, beta, Lambda, w, model)
```

Arguments

`Time`	right censored data which is the follow up time.
`Status`	the censoring indicator, normally 1 = event of interest happens, and 0 = censoring.
`X`	a matrix of covariates corresponding to the latency part.
`beta`	initial beta from the GEE for the latency part.
`Lambda`	initial cumulative baseline hazard function from the GEE with independence working corrlation matrix.
`w`	conditional probability of a patient remains uncured at the mth iteration. We use Status as initial value.
`model`	specifies your model, it can be `para` which represents the parametric PHMC model with two-parameter Weibull baseline survival function, or `semi` which represents the semiparametric PHMC model.

References

Kalbfleisch, J. D. and Prentice, R. L. (2002) *The Statistical Analysis of Failure Time Data*. John Wiley & Sons, New York, 2nd edition.

bmt *Bone marrow transplantation data*

Description

This multi-center acute leukemia study consists of 137 patients with acute myelocytic leukemia (AML) or acute lymphoblastic leukemia (ALL) aged 7 to 52 from March 1, 1984 to June 30, 1989 at four institutions (Klein and Moeschberger, 2003). The failure time on study is defined at time (in days) to relapse or death.

Usage

```
data(bmt)
```

Format

The variables represented in the data set are as follows:

`g` Disease group: 1 - All, 2 - AML Low Risk, 3 - AML High Risk.

`T1` Time to death or on study time.

`T2` Disease free survival time (time to relapse, death or end of study).

`d1` Death indicator: 1 - Dead, 0 - Alive.

`d2` Relapse indicator: 1 - Relapsed, 0 - Disease Free.

`d3` Disease free survival indicator: 1 - Dead or Relapsed, 0 - Alive Disease Free.

`TA` Time to Acute Graft-Versus-Host Disease.

`da` Acute GVHD indicator: 1 - Developed Acute GVHD, 0 - Never Developed Acute GVHD.

`TC` Time to Chronic Graft-Versus-Host Disease.

`dc` Chronic GVHD Indicator: 1 - Developed Chronic GVHD, 0 - Never Developed Chronic GVHD.

`TP` Time to return of platelets to normal levels.

`dp` Platelet recovery indicator: 1 - platelets returned to normal, 0 - platelets never returned to normal.

`Z1` Patient age in years.

`Z2` Donor age in years.

`Z3` Patient sex: 1 - Male, 0 - Female.

`Z4` Doner sex: 1 - Male, 0 - Female.

`Z5` Patient CMV status: 1 - CMV positive, 0 - CMV negative.

`Z6` Donor CMV status: 1 - CMV positive, 0 - CMV negative.

`Z7` Waiting time to transplant in days.

`Z8` FAB: 1 - FAB Grade 4 or 5 and AML, 0 - otherwise.

`Z9` Hospital: 1 - The Ohio State University, 2 - Alferd , 3 - St. Vincent, 4 - Hahnemann.

`Z10` MTX: used as a Graft-Versus-Host-Prophylactic 1 - Yes, 0 - No.

References

Klein, J. P. and Moeschberger, M. L. (2003) *Survival Analysis: Techniques for Censored and Truncated Data*. Springer, New York, 2nd edition.

`emes` *Expectation-Maximization (EM) algorithm and Expectation-Solution (ES) algorithm*

Description

EM algorithm is based on Peng et al. (2007) and ES algorithm is based on Niu and Peng (2013). ES algorithm is an estension of the EM algorithm where the M-step of the EM algorithm is replaced by a step requiring the solution of a series of generalised estimating equations. Both algorithm are used for the analysis of survival cure data with potential correlation.

Usage

```
emes(Time, Status, X, Z, id, corstr, stdz, esmax, eps)
```

Arguments

`Time` right censored data which is the follow up time.

`Status` the censoring indicator, normally 0 = event of interest happens, and 0 = censoring.

`X` a matrix of covariates corresponding to the latency part.

`Z` a matrix of covariates corresponding to the incidence part.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`corstr` a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

`stdz` If it is TRUE, all the covariates in the `formula` and `cureform` are standardized. By default, `stdz = FALSE`.

`esmax` specifies the maximum iteration number. If the convergence criterion is not met, the ES iteration will be stopped after `esmax` iterations and the estimates will be based on the last ES iteration. The default `esmax = 100`.

`eps` tolerance for convergence. The default is `eps = 1e-6`. Iteration stops once the relative change in deviance is less than `eps`.

es *Expectation-Solution (ES) algorithm*

Description

ES algorithm is an estension of the EM algorithm where the M-step of the EM algorithm is replaced by a step requiring the solution of a series of generalised estimating equations. We use the ES algorithm for the analysis of survival cure data with potential correlation.

Usage

```
es(Time, Status, X, Z, id, model, corstr, stdz, esmax, eps)
```

Arguments

`Time` right censored data which is the follow up time.

`Status` the censoring indicator, normally 0 = event of interest happens, and 0 = censoring.

`X` a matrix of covariates corresponding to the latency part.

`Z` a matrix of covariates corresponding to the incidence part.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`model` specifies your model, it can be `para` which represents the parametric PHMC model with two-parameter Weibull baseline survival function, or `semi` which represents the semiparametric PHMC model.

`corstr` a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

`stdz` If it is TRUE, all the covariates in the `formula` and `cureform` are standardized. By default, `stdz = FALSE`.

`esmax` specifies the maximum iteration number. If the convergence criterion is not met, the ES iteration will be stopped after `esmax` iterations and the estimates will be based on the last ES iteration. The default `esmax = 100`.

`eps` tolerance for convergence. The default is `eps = 1e-6`. Iteration stops once the relative change in deviance is less than `eps`.

geebt *Generalized estimating equations for the latency part*

Description

Fit the PH model in the latency part with the GEE approach.

Usage

```
geebt(Status, Lambda, X, beta, w, id, corstr)
```

Arguments

`Status` the censoring indicator, normally 0 = event of interest happens, and 0 = censoring

`Lambda` initial cumulative baseline hazard function from the GEE with independence working corrlation matrix.

`X` a matrix of covariates corresponding to the latency part.

`beta` initial beta for the GEE for the latency part. We use 0 as the initial value.

`w` conditional probability of a patient remains uncured at the mth iteration. We use Status as initial value.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`corstr` a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

geecure *Marginal proportional hazards mixture cure model with generalzied estimating equations*

Description

Fit the marginal proportional hazards mixture cure (PHMC) model with the generalized estimating equations (GEE). GEE approach is generalized to the marginal PHMC model through the expectation-solution (ES) algorithm to account for the correlation among the cure statuses and the dependence among the failure times of uncured patients to improve the estimation efficiency.

Usage

```
geecure(formula, cureform, data, id, model = c("para", "semi"),
        corstr = c("independence", "exchangeable"), Var = TRUE, stdz = FALSE,
        boots = FALSE, nboot = 100, esmax = 100, eps = 1e-06)
```

Arguments

`formula` a formula expression, of the form `response ~ predictors`. The `response` is a `Surv` object with right censoring. It is used to specify the covariate effects on the failure time of uncured subjects. See the documentation for survreg, Surv for details. The expression to the right of the "~" specifies the effect of covariates on the failure time of uncured patients.

`cureform` a formula expression, of the form `cureform ~ predictors`. It is used to specify the effects of covariates on the cure rate. A covariate may be used in both `formula` and `cureform`.

`data` a data frame in which to interpret the variables named in the `formula` and the `cureform`.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`model` specifies your model, it can be `para` which represents the parametric PHMC model with two-parameter Weibull baseline survival function, or `semi` which represents the semiparametric PHMC model.

`corstr` a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

`Var` If it is TRUE, the program returns Std.Error by the sandwich method. By default, `Var = TRUE`.

`stdz` If it is TRUE, all the covariates in the `formula` and `cureform` are standardized. By default, `stdz = FALSE`.

`boots` If it is TRUE, the program returns Std.Error by the bootstrap method. By default, `boots = FALSE`.

`nboot` the number of bootstrap samples. The default is `nboot = 100`.

`esmax` specifies the maximum iteration number. If the convergence criterion is not met, the ES iteration will be stopped after `esmax` iterations and the estimates will be based on the last ES iteration. The default `esmax = 100`.

`eps` tolerance for convergence. The default is `eps = 1e-6`. Iteration stops once the relative change in deviance is less than `eps`.

Details

The marginal PHMC model is considered in this function. For cure rate, a logistic regression model is employed and the probability of being cured is given by $(1+\exp(\gamma Z))^{(-1)}$. For uncured subject, the failure time is modeled by either the parametric proportional hazards model with Weibull baseline distributions or the semiparametric proportional hazards model. A covariate can be used either in `formula` or in `cureform` or in both. The model parameters are estimated by the expectation-solution (ES) algorithm and the standard error estimates are obtained from sandwich variance formula based on Niu and Peng (2014) and Niu et al. (2018).

Value

An object of class `geecure` is returned. It can be examined by `print.geecure()`.

References

Niu, Y. and Peng, Y. (2014) Marginal regression analysis of clustered failure time data with a cure fraction. *Journal of Multivariate Analysis*, **123**: 129-142.

Niu, Y., Song, L., Liu, Y, and Peng, Y. (2018) Modeling clustered long-term survivors using marginal mixture cure model. *Biometrical Journal*, doi: 10.1002/bjmj.201700114.

Examples

```
# Be patient, the following examples may take several minites on a faster computer.
# Example 1. Fit the marginal parametric PHMC model for the smoking cessation data.
data(smoking)
smoking$Time <- ifelse(smoking$Relapse == 0, smoking$Timept1,
                       (smoking$Timept1 + smoking$Timept2)/2)

# plot the KM survival curve of smoking cessation data
plot(survfit(Surv(Time, Relapse) ~ SexF + (SI.UC), data = smoking),
      ylab = "Survival function", xlab = "Years", ylim = c(0.5, 1),
      xlim = c(0, 6), lty = 1:4, col = 1:4)
legend(0.5, 0.63, c("SI/Male", "SI/Female", "UC/Female", "UC/Male"), cex = 1,
        col = c(2, 4, 3, 1), lty = c(2, 4, 3, 1))

geesmokingind <- geecure(Surv(Time, Relapse) ~ SexF + Duration + SI.UC + F10Cigs +
      SexF * SI.UC, cureform = ~ SexF + Duration + SI.UC + F10Cigs + SexF * SI.UC,
      data = smoking,  model = "para", id = smoking$Zip, corstr = "independence")

geesmokingexch <- geecure(Surv(Time, Relapse) ~ SexF + Duration + SI.UC + F10Cigs +
      SexF * SI.UC, cureform = ~ SexF + Duration + SI.UC + F10Cigs + SexF * SI.UC,
      data = smoking,  model = "para", id = smoking$Zip, corstr = "exchangeable")

# Example 2. Fit the marginal semiparametric PHMC model for the bmt data.
data(bmt)
bmt$g <- factor(bmt$g, label = c("ALL", "AML low risk", "AML high risk"))
bmt$Z8 <- factor(bmt$Z8, label = c("Otherwise", "FAB"))

# plot the KM survival curve of bmt data
plot(survfit(Surv(T2, d3) ~ 1, data = bmt), xlab = "Days", ylab = "Survival Probability",
     cex.lab = 1.7, cex.axis = 2, cex.main = 1.7, mark.time = TRUE)

geebmtind <- geecure(Surv(T2, d3) ~ factor(g) + Z8, cureform = ~ factor(g) + Z8,
             data = bmt, model = "semi", id = bmt$Z9, corstr= "independence")

geebmtexch <- geecure(Surv(T2, d3) ~ factor(g) + Z8, cureform = ~ factor(g) + Z8,
              data = bmt, model = "semi", id = bmt$Z9, corstr= "exchangeable",
              stdz = TRUE, boots = TRUE)

# Example 3. Fit the marginal semiparametric PHMC model for the tonsil data.
data(tonsil)
tonsil<-tonsil[-c(141,136,159),]
tonsil$Sex <- ifelse(tonsil$Sex == 1, 0, 1)
tonsil$Cond <- ifelse(tonsil$Cond == 1, 0, 1)
tonsil$T <- ifelse(tonsil$T < 4, 0, 1)
```

```
# plot the KM survival curve of tonsil data
plot(survfit(Surv(Time, Status) ~ 1, data = tonsil), xlab = "Days", ylab = "Survival
     Probability", cex.lab = 1.7, cex.axis = 2, cex.main = 1.7, mark.time = TRUE)

geetonsilind <- geecure2(Surv(Time, Status) ~ Sex + factor(Grade) + Age + Cond + T,
                cureform = ~ Sex + factor(Grade) + Age + Cond + T, data = tonsil,
                id = tonsil$Inst, corstr = "independence")

geetonsilexch <- geecure2(Surv(Time, Status) ~ Sex + factor(Grade) +Age + Cond + T,
                 cureform = ~ Sex + factor(Grade) + Age + Cond + T, data = tonsil,
                id = tonsil$Inst, corstr = "exchangeable",  stdz = TRUE, Var = FALSE)
```

geecure2 *Semiparametric marginal proportional hazards mixture cure model*

Description

Fit the semiparametric marginal proportional hazards mixture cure (PHMC) model for clustered failure time data. The function is based on the methods proposed by Peng et al. (2007) and Niu and Peng (2013).

Usage

```
geecure2(formula, cureform, data, id, corstr = c("independence", "exchangeable"),
Var = TRUE, stdz = FALSE, boots = FALSE, nboot = 100, esmax = 100, eps = 1e-06)
```

Arguments

`formula` a formula expression, of the form `response ~ predictors`. The `response` is a `Surv` object with right censoring. It is used to specify the covariate effects on the failure time of uncured subjects. See the documentation for survreg, Surv for details. The expression to the right of the "~" specifies the effect of covariates on the failure time of uncured patients.

`cureform` a formula expression, of the form `cureform ~ predictors`. It is used to specify the effects of covariates on the cure rate. A covariate may be used in both `formula` and `cureform`.

`data` a data frame in which to interpret the variables named in the `formula` and the `cureform`.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`corstr` a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

`Var` If it is TRUE, the program returns Std.Error by the sandwich method. By default, `Var = TRUE`.

`stdz` If it is TRUE, all the covariates in the `formula` and `cureform` are standardized. By default, `stdz = FALSE`.

`boots` If it is TRUE, the program returns Std.Error by the bootstrap method. By default, `boots = FALSE`.

`nboot` the number of bootstrap samples. The default is `nboot = 100`.

`esmax` specifies the maximum iteration number. If the convergence criterion is not met, the ES iteration will be stopped after `esmax` iterations and the estimates will be based on the last ES iteration. The default `esmax = 100`.

`eps` tolerance for convergence. The default is `eps = 1e-6`. Iteration stops once the relative change in deviance is less than `eps`.

Details

The semiparametric marginal PHMC model is considered in this function. For cure rate, a logistic regression model is employed and the probability of being cured is given by $(1+\exp(\gamma Z))^{(-1)}$. For uncured subject, the failure time is modeled by the semiparametric proportional hazards model. A covariate can be used either in `formula` or in `cureform` or in both. When `corstr = independence`, the model parameters are estimated by the expectation-maximization (EM) algorithm and the standard error estimates are obtained from sandwich variance formula based on Peng et al. (2007). When `corstr = exchangeable`, `stdz = TRUE` and `boots = TRUE`, the model parameters are estimated by the expectation-solution (ES) algorithm and the standard error estimates are obtained from bootstrap variance formula based on and Niu et al. (2013).

Value

An object of class `geecure2` is returned. It can be examined by `print.geecure2()`.

References

Peng, Y., Taylor, J. M. G., and Yu, B. (2007) A marginal regression model for multivariate failure time data with a surviving fraction. *Lifetime Data Analysis*, **13**: 351-369.

Niu, Y. and Peng, Y. (2013) A semiparametric marginal mixture cure model for clustered survival data. *Statistics in Medicine*, **32**: 2364-2373.

Examples

```
# Example. Fit the marginal semiparametric PHMC model for the bmt data.
data(bmt)
geebmtind2 <- geecure2(Surv(T2, d3) ~ Z8, cureform = ~ Z8, data = bmt, id = bmt$Z9,
              corstr= "independence")
geebmtexch2 <- geecure2(Surv(T2, d3) ~ Z8, cureform = ~ Z8, data = bmt, id = bmt$Z9,
               corstr= "exchangeable", stdz = TRUE, Var = FALSE)
```

geega *Generalized estimating equations for the incidence part*

Description

Fit the logistic model in the incidence part with the GEE approach

Usage

```
geega(w, Z, gamma, id, corstr)
```

Arguments

`w`	conditional probability of a patient remains uncured at the mth iteration. We use Status as initial value.
`Z`	a matrix of covariates corresponding to the incidence part.
`gamma`	initial beta for the GEE for the latency part. We use 0 as the initial value.
`id`	a vector which identifies the clusters. The length of `id` should be the same as the number of observations.
`corstr`	a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

initial_Lambda *Initial value of the cumulative baseline hazard function*

Description

Obtain the initial value of the cumulative baseline hazard function in the latency part through the GEE with the `independence` working correlation matrix.

Usage

```
initial_Lambda(Time, Status, X, Z, id, model, corstr)
```

Arguments

`Time`	right censored data which is the follow up time.
`Status`	the censoring indicator, normally 0 = event of interest happens, and 0 = censoring.
`X`	a matrix of covariates corresponding to the latency part.
`Z`	a matrix of covariates corresponding to the incidence part.
`id`	a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`model` specifies your model, it can be `para` which represents the parametric PHMC model with two-parameter Weibull baseline survival function, or `semi` which represents the semiparametric PHMC model.

`corstr` a character string specifying the correlation structure. The following are permitted: `independence` and `exchangeable`.

`print.geecure` *Print geecure object*

Description

Output of `geecure` object.

Usage

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

Arguments

`x` an object of `geecure`.

`...` further arguments to be added in the `print.geecure` function.

`print.geecure2` *Print geecure2 object*

Description

Output of `geecure2` object.

Usage

```
## S3 method for class 'geecure2'
print(x, ...)
```

Arguments

`x` an object of `geecure2`.

`...` further arguments to be added in the `print.geecure2` function.

`smoking` *A Smoking Cessation Data*

Description

The original data consist of 223 people enrolled in the study between November 1986 and February 1989 from 51 zip codes in the southeastern corner of Minnesota in the United States (Banerjee and Carlin, 2004). In this study, smokers were randomly assigned to one of two treatment groups: smoking intervention (SI) group or usual care (UC) group. The survival time is defined as the time (in years) required for a failed quitter to resume smoking. The people residing in the area with the same zip code form a cluster and may be spatially correlated due to the shared environment.

Usage

```
data(smoking)
```

Format

Observed covariates include

`SexF` 0 = male, 1 = female.

`Duration` duration as smoker in years.

`SI.UC` intervention type: 1 = smoking intervention (SI), 0 = usual care (UC).

`F10Cigs` the average number of cigarettes smoked per day over the last 10 years (rounded).

`Relapse` 1 = relapse, 0 = no relapse.

`Timept1` the time of study entry.

`Timept2` the time of resume smoking.

`Zip` 51 zip codes in the southeastern corner of Minnesota.

References

Banerjee, S. and Carlin, B. P. (2004) Parametric spatial cure rate models for interval-censored time-to-relapse data. *Biometrics*, **60**: 268-275.

`tonsil` *Multi-Center Clinical Trial of Tonsil Carcinoma*

Description

A tonsil cancer clinical trial study conducted by the Radiation Therapy Oncology Group in the United States. The survival time is defined as the time (in days) from diagnosis to death. In this study, patients in one institution were randomly assigned to one of two treatment groups: radiation therapy alone or radiation therapy together with a chemotherapeutic agent. A part of the data from the study is available in Kalbfleisch and Prentice (2002).

Usage

```
data(tonsil)
```

Format

A part of the data from the study is available in Kalbfleisch and Prentice (2002), which includes times (in days) from diagnosis to death of 195 patients with squamous cell carcinoma of three sites in the oropharynx between 1968 and 1972 in six participating institutions. Other variables include

`Inst` institution code, from 1 to 6, represents six participating institutions

`Sex` 1 = male, 2 = female.

`Trt` treatment: 1 = standard, 2 = test.

`Grade` 1 = well differentiated, 2 = moderately differentiated, 3 = poorly differentiated.

`Age` in years at time of diagnosis.

`Cond` condition: 1 = no disability, 2 = restricted work, 3 = requires assistance with self care, 4 = bed confined.

`Site` 1 = faucial arch, 2 = tonsillar fossa, 3 = posterior pillar, 4 = pharyngeal tongue, 5 = posterior wall.

`T` T staging: 1 = primary tumor measuring 2 cm or less in largest diameter; 2 = primary tumor measuring 2 to 4 cm in largest diameter, minimal infiltration in depth; 3 = primary tumor measuring more than 4 cm; 4 = massive invasive tumor.

`N` N staging: 0 = no clinical evidence of node metastases; 1 = single positive node 3 cm or less in diameter, not fixed; 2 = single positive node more than 3 cm in diameter, not fixed; 3 = multiple positive nodes or fixed positive nodes.

`EntryDate` Date of entry: Day of year and year.

`Status` 0 = censored, 1 = dead.

`Time` in days from day of diagnosis.

References

Kalbfleisch, J. D. and Prentice, R. L. (2002) *The Statistical Analysis of Failure Time Data*. John Wiley & Sons, New York, 2nd edition.

`tonsil_bootsample` *A bootstrap sample for tonsil data*

Description

A bootstrap sample of tonsil data by sampling `Zip` with replacement.

Usage

```
data(tonsil_bootsample)
```

`varest` *Variance estimate with sandwich formula based on the ES algorithm*

Description

Calculate the variance estimates using the sandwich formula based on the ES algorithm.

Usage

```
varest(Time, Status, X, Z, id, gamma, beta, kappa, gphi, gcor, bphi, bcor,
       Lambda, w, model)
```

Arguments

`Time` right censored data which is the follow up time.

`Status` the censoring indicator, normally 0 = event of interest happens, and 0 = censoring.

`X` a matrix of covariates corresponding to the latency part.

`Z` a matrix of covariates corresponding to the incidence part.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`gamma` the estimates for the incidence part.

`beta` the estimates for the latency part.

`kappa` the estimate of the shape parameter in the Weibull baseline hazard function when `model = "para"`.

`gphi` the estimate of the scale parameter ϕ_1 in the GEE for the incidence part.

`gcor` the estimate of the correlation parameter ρ_1 in the GEE for the incidence part.

`bphi` the estimate of the scale parameter ϕ_2 in the GEE for the latency part.

`bcor` the estimate of the correlation parameter ρ_2 in the GEE for the latency part.

`Lambda` the estimate of the cumulative baseline hazard function in the GEE for the latency part.

`w` conditional probability of a patient remains uncured.

`model` specifies your model, it can be `para` which represents the parametric PHMC model with two-parameter Weibull baseline survival function, or `semi` which represents the semiparametric PHMC model.

varest2 — *Variance estimate with sandwich formula based on the EM algorithm*

Description

Calculate the variance estimates using the sandwich formula based on the EM algorithm.

Usage

```
varest2(Time, Status, X, Z, id, gamma, beta, bsurv, w)
```

Arguments

`Time` right censored data which is the follow up time.

`Status` the censoring indicator, normally 0 = event of interest happens, and 0 = censoring.

`X` a matrix of covariates corresponding to the latency part.

`Z` a matrix of covariates corresponding to the incidence part.

`id` a vector which identifies the clusters. The length of `id` should be the same as the number of observations.

`gamma` the estimates for the incidence part.

`beta` the estimates for the latency part.

`bsurv` the estimate of the baseline survival function for the latency part.

`w` conditional probability of a patient remains uncured.

Index