

Package ‘WeightCalibSurvival’

July 29, 2022

Title Weight Calibration for Additive and Cox Hazards Models

Version 1.0.0

Description Using weight calibration to improve efficiency for estimating pure risks from additive and Cox hazards models with two-phase designs such as nested case-control design.

Depends R (>= 3.5.1)

License GPL-2 + file LICENSE

Encoding UTF-8

Imports graphics, stats, MASS, survival, timereg, caret

NeedsCompilation no

Author Yei Eun Shin [aut, cre]

Maintainer Yei Eun Shin <syeeun@gmail.com>

R topics documented:

WeightCalibSurvival-package	1
addHazard	2
addHazardPureRisk	4
control	5
coxHazard	6
coxHazardPureRisk	8
diagnosticTests	9
sample_data	10
Index	12

WeightCalibSurvival-package

Weight Calibration for Additive Hazards and Cox Proportional Hazards Models

Description

To improve efficiency for estimating pure risks from additive hazards and Cox proportional hazards models with nested case-control designs.

Details

Methods in this package are applicable to studies with some covariates that are only available for case-control samples nested in a cohort. More efficient estimates of pure risks are obtained by calibrating the design weights to data available in the entire cohort. Variance estimates for the weight-calibrated estimates are based on explicit formulas developed in the references below.

Author(s)

Yei Eun Shin <syeeun@gmail.com>, Ruth M. Pfeiffer, Barry I. Graubard, and Mitchell H. Gail

References

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve efficiency for estimating pure risks from the additive hazards model with the nested case-control design. *Biometrics*. 2020;1-13. <https://doi.org/10.1111/biom.13413>

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve the efficiency of pure risk estimates from case-control samples nested in a cohort. *Biometrics*. 2019;1-11. <https://doi.org/10.1111/biom.13209>

addHazard

Fit Additive Hazards Model from Nested Case-control Samples (NCC) using weight calibration

Description

This function fits the additive hazards model with time-varying coefficients and time-invariant coefficients from a nested case-control study with or without weight calibration.

Usage

```
addHazard(data, ncc.subset, outcome.var, time.vars, timeDep.covars, timeIndep.covars,
          anc.covars, risk.time, risk.time0=0, nCnt1PerCase=1,
          inclProb.var=NULL, nrisk.var=NULL,
          print=1, min.inclProb=1e-6, control=NULL)
```

Arguments

<code>data</code>	Data frame containing all the data for analysis.
<code>ncc.subset</code>	Logical vector giving the subjects in the nested case-control sample. The length must be equal to the number of rows in <code>data</code> .
<code>outcome.var</code>	Binary outcome variable in <code>data</code> (1 = case, 0 = non-case).
<code>time.vars</code>	One or two time-to-event variables in <code>data</code> .
<code>timeDep.covars</code>	Character vector of covariates with time-varying effects.
<code>timeIndep.covars</code>	Character vector of covariates with time-invariant effects.
<code>anc.covars</code>	NULL or a character vector of ancillary covariates used to impute missing values of <code>timeDep.covars</code> and <code>timeIndep.covars</code> for subjects not in the nested case-control sample. If NULL, then random noise will be used for the imputation (see details).
<code>risk.time</code>	Projection time for pure risk.

<code>risk.time0</code>	Initial projection time for pure risk. Only used if <code>time.vars</code> contains two time variables. The default is 0.
<code>nCnt1PerCase</code>	Number of controls per case. The default is 1.
<code>inclProb.var</code>	NULL or a variable in data giving the probability of being included in the nested case-control sample. This can be computed using the function <code>multipleNCC::KMprob</code> . If NULL, then it is internally computed based on <code>outcome.var</code> , <code>time.vars</code> , and <code>nCnt1PerCase</code> . The default is NULL.
<code>nrisk.var</code>	NULL or a variable in data giving the number of at-risk subjects at each time. If NULL, then it is internally computed based on <code>outcome.var</code> , <code>time.vars</code> , and <code>nCnt1PerCase</code> . The default is NULL.
<code>print</code>	0 or 1 to print information. The default is 1.
<code>min.inclProb</code>	Positive value ≤ 1 giving the minimum inclusion probability for being in the nested case-control sample. Any inclusion probability less than <code>min.inclProb</code> will be set to <code>min.inclProb</code> . This cannot be 0, since weights will be formed by taking the reciprocals of the inclusion probabilities. The default is 1e-6.
<code>control</code>	See control .

Details

NOTE: If `anc.covars = NULL`, then a seed must be set in order to reproduce the results. See Step 1 below.

The algorithm is as follows:

Step 1: Obtain predictions for missing data

For any variable in `timeDep.covars` and `timeIndep.covars` with missing data from phase 1, fit a weighted generalized linear model among the nested case-control subjects (phase 2) with that variable as the outcome adjusting for the other covariates in `timeDep.covars`, `timeIndep.covars` and `anc.covars`. The weights in the model are the reciprocals of the inclusion probabilities (`inclProb.var`). If `anc.covars = NULL`, then `runif(nrow(data))` will be used as an ancillary covariate.

Step 2: Create auxiliary statistics (influence functions)

Step 3: Calibrate the design weights

Step 4: Fit the additive hazards model with and without weight calibration.

Value

A list containing two lists: `with.calibration` and `without.calibration`. Each sublist contains the parameter estimates, standard errors, and other objects needed to estimate pure risk using [addHazardPureRisk](#).

Author(s)

Yei Eun Shin <syeeun@gmail.com>

References

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve efficiency for estimating pure risks from the additive hazards model with the nested case-control design. *Biometrics*. 2020;1-13. <https://doi.org/10.1111/biom.13413>

See Also

[coxHazard](#), [diagnosticTests](#)

Examples

```
data(sample_data, package="WeightCalibSurvival")

# Set the input arguments
ncc.subset <- sample_data[, "ind.ph2"]
outcome.var <- "ind.fail"
time.vars <- "eventime"
timeDep.covars <- c("X1", "X2")
timeIndep.covars <- c("Z1", "Z2")
anc.covars <- "U"
risk.time <- 8
inclProb.var <- "incl.prob"
nrisk.var <- "nrisk"

addHazard(sample_data, ncc.subset, outcome.var, time.vars, timeDep.covars, timeIndep.covars,
  anc.covars, risk.time, inclProb.var=inclProb.var, nrisk.var=nrisk.var)
```

addHazardPureRisk	<i>Additive Hazards Pure Risk</i>
-------------------	-----------------------------------

Description

A pure risk is estimated from additive hazards model fit for a given vector of covariates. Its variance estimate (standard error) is also computed based on influence functions of model coefficients.

Usage

```
addHazardPureRisk(addHazObj, newdata, risk.time,
  new.risk.col="pureRisk", new.risk.se.col="pureRisk.SE")
```

Arguments

addHazObj	A fitted model from addHazard.
newdata	Data frame for covariate-specific pure risk estimation.
risk.time	Projection time for pure risk.
new.risk.col	New column name added to newdata containing the estimates for pure risk. The default is "pureRisk".
new.risk.se.col	New column name added to newdata containing the estimated standard errors for pure risk. The default is "pureRisk.SE".

Details

The covariates used in `addHazObj` must also be in `newdata`, and they must be of the same type.

Value

The data frame `newdata` with two additional columns containing the estimated pure risk and corresponding standard error.

Author(s)

Yei Eun Shin <syeeun@gmail.com>

References

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve efficiency for estimating pure risks from the additive hazards model with the nested case-control design. *Biometrics*. 2020;1-13. <https://doi.org/10.1111/biom.13413>

See Also

[addHazard](#)

Examples

```
data(sample_data, package="WeightCalibSurvival")

# Set the input arguments
ncc.subset <- sample_data[, "ind.ph2"]
outcome.var <- "ind.fail"
time.vars <- "eventime"
timeDep.covars <- c("X1", "X2")
timeIndep.covars <- c("Z1", "Z2")
anc.covars <- "U"
risk.time <- 8
inclProb.var <- "incl.prob"
nrisk.var <- "nrisk"

fit <- addHazard(sample_data, ncc.subset, outcome.var, time.vars, timeDep.covars, timeIndep.covars,
  anc.covars, risk.time, inclProb.var=inclProb.var, nrisk.var=nrisk.var)

newdata <- sample_data[ncc.subset, ]
ret <- addHazardPureRisk(fit$with.calibration, newdata, risk.time)
ret[1:5, ]
```

control

List for controlling weight calibration

Description

List of convergence parameters for the weight calibration algorithm

Format

The format is: List of 3

maxit Maximum number of iterations. The default is 1000.

eps Stopping tolerance. The default is 1e-8.

conv "abs" or "rel" for absolute or relative convergence. The default is "abs".

coxHazard	<i>Fit Cox Model from Nested Case-control Samples (NCC) using weight calibration</i>
-----------	--

Description

This function fits the Cox proportional hazards model from a nested case-control study using weight calibration.

Usage

```
coxHazard(data, ncc.subset, outcome.var, time.vars, covars,
          anc.covars, risk.time, risk.time0=0, nCntlPerCase=1,
          inclProb.var=NULL, print=1, min.inclProb=1e-6, control=NULL)
```

Arguments

data	Data frame containing all the data for analysis.
ncc.subset	Logical vector giving the subjects in the nested case-control sample. The length must be equal to the number of rows in data.
outcome.var	Binary outcome variable in data (1 = case, 0 = non-case).
time.vars	One or two time-to-event variables in data.
covars	Character vector of covariates.
anc.covars	NULL or a character vector of ancillary covariates used to impute missing values of covars for subjects not in the nested case-control sample. If NULL, then random noise will be used for the imputation (see details).
risk.time	Projection time for pure risk.
risk.time0	Initial projection time for pure risk. Only used time.vars contains two time variables. The default is 0.
nCntlPerCase	Number of controls per case. The default is 1.
inclProb.var	NULL or a variable in data giving the probability of being included in the nested case-control sample. This can be computed using the function <code>multipleNCC::KMprob</code> . If NULL, then it is internally computed based on <code>outcome.var</code> , <code>time.vars</code> , and <code>nCntlPerCase</code> . The default is NULL.
print	0 or 1 to print information. The default is 1.
min.inclProb	Positive value ≤ 1 giving the minimum inclusion probability for being in the nested case-control sample. Any inclusion probability less than <code>min.inclProb</code> will be set to <code>min.inclProb</code> . This cannot be 0, since weights will be formed by taking the reciprocals of the inclusion probabilities. The default is 1e-6.
control	See control .

Details

NOTE: If `anc.covars = NULL`, then a seed must be set in order to reproduce the results. See Step 1 below.

The algorithm is as follows:

Step 1: Obtain predictions for missing data

For any variable in `covars` with missing data from phase 1, fit a weighted generalized linear model among the nested case-control subjects (phase 2) with that variable as the outcome adjusting for the other covariates in `covars` and `anc.covars`. The weights in the model are the reciprocals of the inclusion probabilities (`inclProb.var`). If `anc.covars = NULL`, then `runif(nrow(data))` will be used as an ancillary covariate.

Step 2: Create auxiliary statistics (influence functions)

Step 3: Calibrate the design weights

Step 4: Fit the Cox model with and without the calibrated weights.

Value

A list containing two fitted `coxph` models: `with.calibration` and `without.calibration`.

Author(s)

Yei Eun Shin <syeeun@gmail.com>

References

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve the efficiency of pure risk estimates from case-control samples nested in a cohort. *Biometrics*. 2019;1-11. <https://doi.org/10.1111/biom.13209>

See Also

[addHazard](#), [diagnosticTests](#)

Examples

```
data(sample_data, package="WeightCalibSurvival")

# Set the input arguments
ncc.subset <- sample_data[, "ind.ph2"]
outcome.var <- "ind.fail"
time.vars <- "eventime"
covars <- c("X1", "X2", "Z1", "Z2")
anc.covars <- "U"
risk.time <- 8
inclProb.var <- "incl.prob"

coxHazard(sample_data, ncc.subset, outcome.var, time.vars, covars,
           anc.covars, risk.time, inclProb.var=inclProb.var)
```

coxHazardPureRisk *Cox Proportional Hazards Pure Risk*

Description

Estimate pure risk from Cox proportional hazards model given a set of covariates.

Usage

```
coxHazardPureRisk(coxHazObj, newdata, risk.time, risk.time0=0,
                  new.risk.col="pureRisk")
```

Arguments

coxHazObj	A fitted model from addHazard.
newdata	Data frame for covariate-specific pure risk estimation.
risk.time	Projection time for pure risk.
risk.time0	Initial projection time for pure risk. The default is 0.
new.risk.col	New column name added to newdata containing the estimates for pure risk. The default is "pureRisk".

Details

The covariates used in coxHazObj must also be in newdata, and they must be of the same type.

Value

The data frame newdata with one additional column containing the estimated pure risk.

Author(s)

Yei Eun Shin <syeeun@gmail.com>

References

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve efficiency for estimating pure risks from the additive hazards model with the nested case-control design. *Biometrics*. 2020;1-13. <https://doi.org/10.1111/biom.13413>

See Also

[coxHazard](#)

Examples

```
data(sample_data, package="WeightCalibSurvival")

# Set the input arguments
ncc.subset <- sample_data[, "ind.ph2"]
outcome.var <- "ind.fail"
time.vars <- "eventime"
covars <- c("X1", "X2", "Z1", "Z2")
```

```

anc.covars      <- "U"
risk.time      <- 8
inclProb.var    <- "incl.prob"

fit <- coxHazard(sample_data, ncc.subset, outcome.var, time.vars, covars,
                 anc.covars, risk.time, inclProb.var=inclProb.var)

newdata <- sample_data[ncc.subset, ]
ret     <- coxHazardPureRisk(fit$with.calibration, newdata, risk.time)
ret[1:5, ]

```

diagnosticTests *Diagnostic Tests for Additive Hazards and Cox Proportional Hazards*

Description

Perform diagnostic tests for an additive hazard model and the proportional hazards assumption.

Usage

```

diagnosticTests(data, ncc.subset, outcome.var, time.vars,
               timeDep.covars, timeIndep.covars, nCntlPerCase=1,
               inclProb.var=NULL, start.time=0, print=TRUE, plot=TRUE,
               min.inclProb=1e-6)

```

Arguments

data	Data frame containing all the data for analysis.
ncc.subset	Logical vector giving the subjects in the nested case-control sample. The length must be equal to the number of rows in data.
outcome.var	Binary outcome variable in data (1 = case, 0 = non-case).
time.vars	One or two time-to-event variables in data.
timeDep.covars	Character vector of covariates with time-varying effects.
timeIndep.covars	Character vector of covariates with time-invariant effects.
nCntlPerCase	Number of controls per case. This is only used if inclProb.var = NULL. The default is 1.
inclProb.var	NULL or a variable in data giving the probability of being included in the nested case-control sample. This can be computed using the function <code>multipleNCC::KMprob</code> . If NULL, then it is internally computed based on <code>outcome.var</code> , <code>time.vars</code> , and <code>nCntlPerCase</code> . The default is NULL.
start.time	The starting time used to test for constant effects. The default is 0.
print	TRUE or FALSE to print information. The default is TRUE.
plot	TRUE or FALSE to create plots. The default is TRUE.
min.inclProb	Positive value ≤ 1 giving the minimum inclusion probability for being in the nested case-control sample. Any inclusion probability less than <code>min.inclProb</code> will be set to <code>min.inclProb</code> . This cannot be 0, since weights will be formed by taking the reciprocals of the inclusion probabilities. The default is 1e-6.

Details

To test for proportional hazards, the `cox.zph` function is used. To determine if an additive hazards model is appropriate, the `aalen` function is used to perform a Kolmogorov-Smirnov test for time-varying effects.

Value

A list containing two tables: `propHaz.test` and `addHaz.test`.

Author(s)

Yei Eun Shin <syeeun@gmail.com>

References

Shin YE, Pfeiffer RM, Graubard BI, Gail MH. Weight calibration to improve efficiency for estimating pure risks from the additive hazards model with the nested case-control design. *Biometrics*. 2020;1-13. <https://doi.org/10.1111/biom.13413>

See Also

[addHazard](#), [coxHazard](#)

Examples

```
data(sample_data, package="WeightCalibSurvival")

# Set the input arguments
ncc.subset      <- sample_data[, "ind.ph2"]
outcome.var     <- "ind.fail"
time.vars       <- "eventime"
timeDep.covars  <- c("X1", "X2")
timeIndep.covars <- c("Z1", "Z2")
inclProb.var    <- "incl.prob"

ret <- diagnosticTests(sample_data, ncc.subset, outcome.var, time.vars,
                       timeDep.covars, timeIndep.covars, inclProb.var=inclProb.var)
print(ret)
```

sample_data

Sample Data for examples

Description

Sample cohort with simulated nested case-control study where $N = 1000$ and 173 cases with $m = 1$ control per case. This dataset is used to run the provided examples in this package.

Usage

```
sample_data
```

Format

A data frame with N = 1000 rows and 10 variables:

eventime time-to-event (event = either case or censored)

ind.fail binary outcome status (1 = case; 0 = non-case)

X1 binary model covariates with time-varying effects; available in a full cohort

X2 binary model covariates with time-varying effects; available in a full cohort

Z1 binary model covariates with time-invariant effects; available in a full cohort

Z2 binary model covariates with time-invariant effects; only available in nested case-control samples; NA if missing

U an ancillary predictor for Z2

ind.ph2 phase 2 inclusion status (1 = included as a case or selected control; 0 = otherwise)

nrisk number of at-risk subjects at each 'eventime'

incl.prob inclusion probability

Examples

```
data(sample_data, package="WeightCalibSurvival")
```

```
sample_data[1:5, ]
```

Index

- * **datasets**
 - sample_data, [10](#)
- * **misc**
 - control, [5](#)
- * **model, Cox**
 - coxHazard, [6](#)
- * **model, addHazard**
 - addHazard, [2](#)
- * **package**
 - WeightCalibSurvival-package, [1](#)
- * **risk, addHazard**
 - addHazardPureRisk, [4](#)
- * **risk, cox**
 - coxHazardPureRisk, [8](#)
- * **test**
 - diagnosticTests, [9](#)

aalen, [10](#)
addHazard, [2](#), [5](#), [7](#), [10](#)
addHazardPureRisk, [3](#), [4](#)

control, [3](#), [5](#), [6](#)
cox.zph, [10](#)
coxHazard, [4](#), [6](#), [8](#), [10](#)
coxHazardPureRisk, [8](#)
coxph, [7](#)

diagnosticTests, [4](#), [7](#), [9](#)

sample_data, [10](#)

WeightCalibSurvival
(WeightCalibSurvival-package),
[1](#)
WeightCalibSurvival-package, [1](#)