Package: TwoPhaseInd (via r-universe)

August 26, 2024

Type Package

Title Estimate Gene-Treatment Interaction Exploiting Randomization

Version 1.1.2

Author James Dai [aut, cre], Xiaoyu Wang [aut]

Maintainer James Dai <jdai@fredhutch.org>

Description Estimation of gene-treatment interactions in randomized clinical trials exploiting gene-treatment independence. Methods used in the package refer to J. Y. Dai, M. LeBlanc, and C. Kooperberg (2009) Biometrics <doi:10.1111/j.1541-0420.2008.01046.x>.

License GPL (>= 2)

LazyLoad no

NeedsCompilation yes

Imports survival

Repository CRAN

Date/Publication 2022-02-17 00:50:11 UTC

Contents

aco1arm *A function to estimate parameters in augmented case-only designs, the genotype is ascertained for a random subcohort from the active treatment arm or the placebo arm.*

Description

This function estimates parameters of proportional hazards model with gene-treatment interaction. It employs case-cohort estimation incorporating the case-only estimators. The method was published in Dai et al. (2016) Biometrics.

Usage

```
aco1arm(data, svtime, event, treatment, BaselineMarker, subcohort, esttype = 1,
augment = 1, extra)
```

Arguments

data	A data frame used to access the following data.
svtime	A character string of column name, corresponds to one column of the data frame, which is used to store the failure time variable (numeric).
event	A character string of column name, corresponds to one column of the data frame, which is used to store the indicator of failure event (1: failure, 0: not failure).
treatment	A character string of column name, corresponds to one column of the data frame, which is used to store the binary vector of treatment variable (1: treatment, 0: placebo).
BaselineMarker	A character string of column name, corresponds to one column of the data frame, which is used to store a vector of biomarker.
subcohort	A character string of column name, corresponds to one column of the data frame, which is used to store the indicator of sub-cohort (1: sample belong to the sub-cohort, 0: not belong to the sub-cohort)
esttype	The option of estimation methods (1: Self-Prentice estimator, 0: Lin-Ying estimator).
augment	The indicator of whether subcohort was drawn from the active treatment arm (augment=1) or from the placebo arm (augment=0).
extra	A string vector of column name(s), corresponds to more or more column(s) of the data frame, which is/are used to store the extra baseline covariate(s) to be adjusted for in addition to treatment and biomarker.

Details

The function returns estimates of the proportional hazards model, and variance of the estimates. The method was published in Dai et al. (2016) Biometrics.

Value

A list of estimates and variance of the estimates.

`Estimate`	A data frame of beta(Estimated parameter), stder(Standard error),and pVal(p value)
`Covariance`	covariance data frame of genotype,treatment,and interaction

Author(s)

James Y. Dai

References

J. Y. Dai, X. C. Zhang,C. Y. Wang, and C. Kooperberg. Augmented case-only designs for randomized clinical trials with failure time endpoints. Biometrics, DOI: 10.1111/biom.12392, 2016.

See Also

`aco2arm`

Examples

```
## Load the example data
data(acodata)

## Augmented data in the active arm

rfit1 <- aco1arm(data=acodata,
                 svtime="vacc1_evinf",
                 event="f_evinf",
                 treatment="f_treat",
                 BaselineMarker="fcgr2a.3",
                 subcohort="subcoh",
                 esttype=1,
                 augment=1,
                 extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm","any_drug",
                         "num_male_part_cat","uias","uras"))

rfit1

## Augmented data in the placebo arm

rfit2 <- aco1arm(data=acodata,
                 svtime="vacc1_evinf",
                 event="f_evinf",
                 treatment="f_treat",
                 BaselineMarker="fcgr2a.3",
                 subcohort="subcoh",
                 esttype=1,
                 augment=0,
                 extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm",
```

```
                        "any_drug","num_male_part_cat","uias","uras"))
rfit2
```

---

aco2arm *A function to estimate parameters in Cox proportional hazards model using augmented case-only designs, the genotype is ascertained for a random subcohort from both the active treatment arm and the placebo arm (case-cohort sampling) or a case-control sample in both arms.*

Description

This function estimates parameters of proportional hazards model with gene-treatment interaction. It employs case-cohort estimation incorporating the case-only estimators. The method was published in Dai et al. (2015) Biometrics.

Usage

```
aco2arm(data, svtime, event, treatment, BaselineMarker, subcohort=NULL,
esttype = NULL, weight=NULL, extra=NULL)
```

Arguments

data	A data frame used to access the following data.
svtime	A character string of column name, corresponds to one column of the data frame, which is used to store the failure time variable (numeric).
event	A character string of column name, corresponds to one column of the data frame, which is used to store the indicator of failure event (1: failure, 0: not failure).
treatment	A character string of column name, corresponds to one column of the data frame, which is used to store the binary vector of treatment variable (1: treatment, 0: placebo).
BaselineMarker	A character string of column name, corresponds to one column of the data frame, which is used to store a vector of biomarker.
subcohort	A character string of column name, corresponds to one column of the data frame, which is used to store the indicator of sub-cohort in the case-cohort sampling (1: sample belong to the sub-cohort, 0: not belong to the sub-cohort). In case-control sampling, this variable is set to be NULL.
esttype	The option of estimation methods (1: Self-Prentice estimator, 0: Lin-Ying estimator).
weight	If the genotype data are obtained through case-control sampling, weight is a vector of sampling weights (inverse of sampling probability) corresponding to rows of data. If the genotype data are obtained through case-cohort sampling, weight is NULL. If a vector of weights have been supplied by user, then esttype is automatically set to 0: Lin-Ying estimator.
extra	A string vector of column name(s), corresponds to more or more column(s) of the data frame, which is/are used to store the extra baseline covariate(s) to be adjusted for in addition to treatment and biomarker.

Details

The function returns estimates of the proportional hazards model, and variance of the estimates. The method was published in Dai et al. (2016) Biometrics.

Value

A list of estimates and variance of the estimates.

`Estimate`	A data frame of beta(Estimated parameter), stder(Standard error),and pVal(p value)
`Covariance`	covariance data frame of genotype,treatment,and interaction

Author(s)

James Y. Dai

References

J. Y. Dai, X. C. Zhang,C. Y. Wang, and C. Kooperberg. Augmented case-only designs for randomized clinical trials with failure time endpoints. Biometrics, DOI: 10.1111/biom.12392, 2016.

See Also

`aco1arm`

Examples

```
## Load the example data
data(acodata)
## Case-cohort + case-only estimators
rfit1 <- aco2arm(data=acodata,
                 svtime="vacc1_evinf",
                 event="f_evinf",
                 treatment="f_treat",
                 BaselineMarker="fcgr2a.3",
                 subcohort="subcoh",
                 esttype=1,
                 weight=NULL,
                 extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm","any_drug",
                 "num_male_part_cat","uias","uras"))
rfit1
```

acoarm *A function to estimate parameters in Cox proportional hazard models by augmented case-only designs for randomized clinical trials with failure time endpoints.*

Description

This function estimates parameters of proportional hazards models with gene-treatment interactions. It employs classical case-cohort estimation methods, incorporating the case-only estimators. The method was published in Dai et al. (2016) Biometrics.

Usage

```
acoarm(data, svtime, event, treatment, BaselineMarker,subcohort, esttype = 1,
augment = 1, weight=NULL, extra = NULL)
```

Arguments

data	A data frame used to access the following data.
svtime	A character string of column name, corresponds to one column of the data frame, which is used to store the failure time variable (numeric).
event	A character string of column name, corresponds to one column of the data frame, which is used to store the indicator of failure event (1: failure, 0: not failure).
treatment	A character string of column name, corresponds to one column of the data frame, which is used to store the binary vector of treatment variable (1: treatment, 0: placebo).
BaselineMarker	A character string of column name, corresponds to one column of the data frame, which is used to store a vector of baseline biomarker that is under investigation for interaction with treatment. The BaselineMarker variable is missing for those who are not sampled in the case-cohort.
subcohort	A character string of column name, corresponds to one column of the data frame, which is used to store the indicator of sub-cohort (1: sample belong to the sub-cohort, 0: not belong to the sub-cohort)
esttype	The option of estimation methods (1: Self-Prentice estimator, 0: Lin-Ying estimator).
augment	The indicator of whether subcohort was drawn from the placebo arm (augment=0), from the active treatment arm (augment=1), or from both arms (augment=2).
weight	If the genotype data are obtained through case-control sampling, weight is a vector of sampling weights (inverse of sampling probability) corresponding to rows of data. If the genotype data are obtained through case-cohort sampling, weight is NULL. If a vector of weights have been supplied by user, then esttype is automatically set to 0: Lin-Ying estimator.
extra	A string vector of column name(s), corresponds to more or more column(s) of the data frame, which is/are used to store the extra baseline covariate(s) to be adjusted for in addition to treatment and biomarker.

Details

The function returns point estimates and standard error estimates of parameters in the proportional hazards model. The method was published in Dai et al. (2015) Biometrics.

Value

beta	Estimated parameter
stder	Estimated standard error of parameter estimates
pVal	p value

Author(s)

James Y. Dai

References

J. Y. Dai, X. C. Zhang,C. Y. Wang, and C. Kooperberg. Augmented case-only designs for randomized clinical trials with failure time endpoints. Biometrics, DOI: 10.1111/biom.12392, 2016.

Examples

```
## Load the example data
data(acodata)
## ACO in placebo arm
rfit0 <- acoarm(data=acodata,
                svtime="vacc1_evinf",
                event="f_evinf",
                treatment="f_treat",
                BaselineMarker="fcgr2a.3",
                subcohort="subcoh",
                esttype=1,
                augment=0,
                weight=NULL,
                extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm",
                "any_drug","num_male_part_cat","uias","uras"))
rfit0

## ACO in active arm
rfit1 <- acoarm(data=acodata,
                svtime="vacc1_evinf",
                event="f_evinf",
                treatment="f_treat",
                BaselineMarker="fcgr2a.3",
                subcohort="subcoh",
                esttype=1,
                augment=1,
                weight=NULL,
                extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm",
                "any_drug","num_male_part_cat","uias","uras"))
rfit1
```

```
## ACO in both arms
rfit2 <- acoarm(data=acodata,
                svtime="vacc1_evinf",
                event="f_evinf",
                treatment="f_treat",
                BaselineMarker="fcgr2a.3",
                subcohort="subcoh",
                esttype=1,
                augment=2,
                weight=NULL,
                extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm",
                "any_drug","num_male_part_cat","uias","uras"))
rfit2
```

---

acodata *A dataset from the STEP trial to study the interactions between gene and vaccine on HIV infection*

Description

A dataset from the STEP trial to study the interactions between gene and vaccine on HIV infection

Usage

```
data("acodata")
```

Format

A data frame with 907 observations on the following 14 variables.

`vacc1_evinf` the time to HIV infection, a numeric vector

`f_evinf` the indicator variable for HIV infection, a numeric vector

`subcoh` the indicator of whether the participant was selected into the sub-cohort for genotyping, a logical vector

`ptid` patricipant identifier, a numeric vector

`f_treat` vaccine assignment variable, a numeric vector

`fcgr2a.3` the genotype of Fcr receptor FcrRIIIa, the biomarker of interest here, a numeric vector

`f_agele30` a numeric vector

`f_hsv_2` a numeric vector

`f_ad5gt18` a numeric vector

`f_crcm` a numeric vector

`any_drug` a numeric vector

`num_male_part_cat` a numeric vector

`uias` a numeric vector

`uras` a numeric vector

Details

A dataset from the STEP trial to study the interactions between gene and vaccine on HIV infection

References

S. P. Buchbinder, D. V. Mehrotra, and D. Ann et al. Efficacy assessment of a cell-mediated immunity HIV-1 vaccine (the Step Study): a double-blind, randomised, placebo-controlled, test-of-concept trial. Lancet. 372(9653):1881-1893, 2008.

J. P. Pandey, A. M. Namboodiri, and S. Bu et l. Immunoglobulin genes and the acquisition of HIV infection in a randomized trial of recombinant adenovirus HIV vaccine. Virology, 441:70-74, 2013.

Examples

```
data(acodata)
## maybe str(acodata)
```

`caseonly` *A function to deal with case-only designs*

Description

This function estimates parameters of case-only designs.

Usage

```
caseonly(data, treatment, BaselineMarker, extra = NULL, fraction = 0.5)
```

Arguments

`data`	A data frame used to access the following data.
`treatment`	A character string of column name, corresponds to one column of the data frame, which is used to store the binary vector of treatment variable (1: treatment, 0: placebo).
`BaselineMarker`	A character string of column name, corresponds to one column of the data frame, which is used to store a vector of biomarker.
`extra`	A string vector of column name(s), corresponds to more or more column(s) of the data frame, which is/are used to store the extra baseline covariate(s) to be included in case-only regression. Note that extra covariates are not needed unless the interactions of treatment and extra coviarates are of interest.
`fraction`	The randomization fraction of active treatment assignment.

Details

This function estimates parameters of case-only designs. It estimates two parameters for "treatment effect when baselineMarker=0"" and treatment+baselineMarker interaction".

Value

For each paramter, it returns:

beta	Estimated parameter
stder	Standard error
pVal	p value

Author(s)

James Y. Dai

References

J. Y. Dai, S. S. Li, and P. B. Gilbert. Case-only methods for competing risks models with application to assessing differential vaccine efficacy by viral and host genetics. Biometrics, 15(1):196-203, 2014.

Examples

```
#form the data
data(acodata)
cdata=acodata[acodata[,2]==1,]
cfit=caseonly(data=cdata,
              treatment="f_treat",
              BaselineMarker="fcgr2a.3",
              extra=c("f_agele30","f_hsv_2","f_ad5gt18","f_crcm",
              "any_drug","num_male_part_cat","uias","uras"))
cfit
```

---

char2num *A function used in acoarm to transform categorical variable to integers*

Description

Transform category data to integers 0..levels(data)-1. The the numeric variable can be then used in acoarm models.

Usage

```
char2num(data)
```

Arguments

data	data is a dataframe composed of categorical variables.

Details

The function transforms a categorical variable to integers.

Value

A data frame of transformed values. For each column, each category is transformed to an integer, from 0 to levels(data[,column])-1.

Author(s)

James Y. Dai

Examples

```
## Load the example data
data(acodata)
result <- char2num(acodata[, "fcgr2a.3"])
```

`mele` *function to compute the maximum estimated likelihood estimator*

Description

This function computes the maximum estimated likelihood estimator (MELE) of regression parameters, which assess treatment-biomarker interactions in studies with two-phase sampling in randomized clinical trials. The function has an option to incorporate the independence between a randomized treatment and the baseline markers.

Usage

```
mele(data, response, treatment, BaselineMarker, extra = NULL, phase,
ind = TRUE, maxit=2000)
```

Arguments

`data`	A data frame used to access the following data. Each row contains the response and predictors of a study participant. All variables are numerical.
`response`	A character string of column name, corresponds to one column of the data frame, which is used to store a numeric vector of response. The response variable should be coded as 1 for cases and 0 for controls.
`treatment`	A character string of column name, corresponds to one column of the data frame, which is used to store a binary vector of the treatment . The treatment variable should be coded as 1 for treatment and 0 for placebo.
`BaselineMarker`	A character string of column name, corresponds to one column of the data frame, which is used to store a vector of biomarker that is assessed for interaction with the treatment. The BaselineMarker variable is missing for those who are not sampled in the second phase.
`extra`	A string vector of column name(s), corresponds to one or more column(s) of the data frame, which are used to store the extra covariate(s) to be adjusted for in addition to treatment and biomarker. All extra variables are missing for those who are not sampled in the second phase.

phase	A character string of column name, correspond to one column of the data frame, which is used to store the indicator of two-phase sampling (1: not being sampled for measuring biomarker; 2: being sampled for measuring biomarker).
ind	A logical flag. TRUE indicates incorporating the independence between the randomized treatment and the baseline markers.
maxit	A integer number of the maximal number of iteration.

Details

The function returns estimates, standard errors, and p values for MELE of a regression model for treatment-biomarker interaction studies with two-phase sampling in randomized trials, response ~ treatment + biomarker + treatment*biomarker + other covariates. Treatment and response are available for all the samples, while baseline biomarker data are available for a subset of samples. The mele can incorporate the independence between the treatment and baseline biomarkers ascertained in the phase-two sample.

Value

beta	Estimated parameter
stder	Standard error
pVal	p value

Author(s)

James Y. Dai

References

J. Y. Dai, M. LeBlanc, and C. Kooperberg. Semiparametric estimation exploiting covariate independence in two-phase randomized trials. Biometrics, 65(1):178-187, 2009.

See Also

spmle

Examples

```
## Load the example data
data(whiBioMarker)
## Here is an example of MELE with exploiting independent and with confounding factors:

melIndExtra <- mele(data=whiBioMarker,  ## dataset
          response="stroke", ## response variable
          treatment="hrtdisp",  ## treatment variable
          BaselineMarker="papbl",  ## environment variable
          extra=c(
             "age"  ## age
                  ## physical activity levels
              , "dias"  ## diabetes
              , "hyp" ## hypertension
```

```
                , "syst"  ## systolic
                , "diabtrt" ## diastolic BP
                , "lmsepi" ## waist:hip ratio
                ), ## extra variable(s)
            phase="phase", ## phase indicator
            ind=TRUE ## independent or non-indepentent
)
```

`remove_missingdata` *A function used in acoarm to remove missing data*

Description

It is used to remove samples which have NA/missing data in covariates.

Usage

```
remove_missingdata(data)
```

Arguments

`data`	data is a dataframe.

Details

The function removes samples (by rows) which have NA/missing data.

Value

A list of the following components.

`idx`	The indices of rows without missing values
`data`	The dataframe without missing values

Author(s)

James Y. Dai

Examples

```
## Load the example data
data(acodata)
result <- remove_missingdata(acodata[, c("vacc1_evinf","fcgr2a.3")])
```

remove_rarevariants *A function used in spmle and acoarm to remove rare-variant covariates*

Description

It is used to remove rare-variant covariates, which can cause divergence problem.

Usage

```
remove_rarevariants(data, cutoff = 0.02)
```

Arguments

data	A dataframe composed of covariates.
cutoff	Proportion cutoff. If data composed of more than (1-cutoff) proportion of a constant value, we call it rare-variant.

Details

The function removes rare-variant covariates.

Value

A logical vector composed of True or False. True means a covariate is rare-variant.

Author(s)

James Y. Dai

Examples

```
## Load the example data
data(acodata)
result <- remove_rarevariants(acodata[, c("vacc1_evinf","fcgr2a.3")])
```

spmle *function to compute the semiparametric maximum likelihood estimator*

Description

This function computes the semiparametric maximum likelihood estimator (SPMLE) of regression parameters, which assess treatment-biomarker interactions in studies with two-phase sampling in randomized clinical trials. The function has an option to incorporate the independence between a randomized treatment and the baseline markers.

Usage

```
spmle(data, response, treatment, BaselineMarker, extra = NULL, phase,
ind = TRUE, difffactor = 0.001, maxit = 1000)
```

Arguments

data	A data frame used to access the following data. Each row contains the response and predictors of a study participant. All variables are numerical.
response	A character string of column name, corresponds to one column of the data frame, which is used to store a numeric vector of response. The response variable should be coded as 1 for cases and 0 for controls.
treatment	A character string of column name, corresponds to one column of the data frame, which is used to store a binary vector of the treatment . The treatment variable should be coded as 1 for treatment and 0 for placebo.
BaselineMarker	A character string of column name, corresponds to one column of the data frame, which is used to store a vector of biomarker that is assessed for interaction with the treatment. The BaselineMarker variable is missing for those who are not sampled in the second phase.
extra	A string vector of column name(s), corresponds to one or more column(s) of the data frame, which are used to store the extra covariate(s) to be adjusted for in addition to treatment and biomarker. All extra variables are missing for those who are not sampled in the second phase.
phase	A character string of column name, correspond to one column of the data frame, which is used to store the indicator of two-phase sampling (1: not being sampled for measuring biomarker; 2: being sampled for measuring biomarker).
ind	A logical flag. TRUE indicates incorporating the independence between the randomized treatment and the baseline markers.
difffactor	A decimal number of the differentiation factor, used to control the step of numerical differentiation.
maxit	A integer number of the maximal number of numerical differentiation iteration.

Details

The function returns estimates, standard errors, and p values for SPMLE for parameters of a regression model for treatment-biomarker interaction studies with two-phase sampling in randomized trials, response ~ treatment + biomarker + treatment*biomarker + other covariates. Treatment and response are available for all the samples, while biomarker data are available for a subset of samples. The SPMLE can incorporate the independence between the treatment and baseline biomarkers ascertained in the phase-two sample. A profile likelihood based Newton-Raphson algorithm is used to compute SPMLE.

Value

beta	Estimated parameter
stder	Standard error
pVal	p value

Author(s)

James Y. Dai

References

J. Y. Dai, M. LeBlanc, and C. Kooperberg. Semiparametric estimation exploiting covariate independence in two-phase randomized trials. Biometrics, 65(1):178-187, 2009.

See Also

mele

Examples

```
## Load the example data
data(whiBioMarker)
## Here is an example of SPMLE with exploiting independent and with confounding factors:
spmleIndExtra <- spmle(data=whiBioMarker,  ## dataset
                       response="stroke",  ## response variable
                       treatment="hrtdisp",  ## treatment variable
                       BaselineMarker="papbl", ## environment variable
                       extra=c(
                         "age"  ## age
                         , "dias" ## diabetes
                         , "hyp" ## hypertension
                         , "syst" ## systolic
                         , "diabtrt" ## diastolic BP
                         , "lmsepi" ## waist:hip ratio
                       ), ## extra variable(s)
                       phase="phase", ## phase indicator
                       ind=TRUE ## independent or non-independent
)
```

---

whiBioMarker *An example dataset to demostrate the usage of MELE and SPMLE*

Description

A dataset from a Women's Health Initiative (WHI) hormone trial to study the interaction between biomarker and hormone therapy on stroke.

Usage

```
data("whiBioMarker")
```

Format

A data frame consisting of 10 observations, with the following columns:

`stroke` a binary indicator vector of stroke; 1=has stroke

`hrtdisp` a binary indicator vector of treatment in the Estrogen Plus Progestin Trial; 1="Estrogen Plus Progestin", 0="placebo"

`papbl` a numeric vector of Biomarker PAP (plasmin-antiplasmin complex) in logarithmic scale (base 10)

`age` an integer vector of age

`dias` A binary indicator vector of Diastolic BP; 1="Yes"

`hyp` a vector of hypertension with levels `Missing`, `No`, `Yes`

`syst` an integer vector of Systolic BP

`diabtrt` A vector of Diabetes with levels: `Missing`, `No`, `Yes`

`lmsepi` A vector of episodes per week of moderate and strenuous recreational physical activity of >= 20 minutes duration with levels `2 - <4 episodes per week`, `4+ episodes per week`, `Missing`, `No activity`, `Some activity`

`phase` a numeric vector of phase; 1: phase 1, 2:phase 2

Details

It is an two-phase sampling example dataset adapted from Kooperberg et al. (2007) to demostrate the usage of MELE and SPMLE algorithms in Dai et al. (2009).

Source

C. Kooperberg, M. Cushman, J. Hsia, J. G. Robinson, A. K. Aragaki, J. K. Lynch, A. E. Baird, K. C. Johnson, L. H. Kuller, S. A. Beresford, and B. Rodriguez. Can biomarkers identify women at increased stroke risk? the women's health initiative hormone trials. PLoS clinical trials, 2(6):e28, Jun 15 2007.

References

J. Y. Dai, M. LeBlanc, and C. Kooperberg. Semiparametric estimation exploiting co-variate independence in two-phase randomized trials. Biometrics, 65(1):178-187, 2009.

Examples

```
data(whiBioMarker)
str(whiBioMarker)
colnames(whiBioMarker)
```

Index