Package ‘Plasmode’

October 15, 2017

Type Package

Title 'Plasmode' Simulation

Version 0.1.0

Author Jessica M. Franklin, Younathan Abdia, Shirley V. Wang.

Maintainer Younathan Abdia <yabdia@bwh.harvard.edu>

Description Creates realistic simulated datasets for causal inference based on a user-supplied example study, Franklin JM, Schneeweiss S, Polinski JM, and Rassen JA (2014) <doi:10.1016/j.csda.2013.10.018>.
It samples units from the data with replacement, and then simulates the exposure, the outcome, or both, based on the observed covariate values in the real data.

License GPL-2

Encoding UTF-8

LazyData true

Imports mgcv(>= 1.8-12), glm2(>= 1.1.2), nlme(>= 3.1-128), survival(>= 2.40-1), MASS, arm, lme4, twang, gbm, lattice, splines, parallel, survey, grid, Matrix, xtable, latticeExtra, RColorBrewer, epiDisplay, foreign, nnet

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-10-15 14:09:58 UTC

R topics documented:

PlasmodeBin *Performs the plasmode simulation*

Description

Creates 'plasmode'simulated datasets based on a given dataset when the outcome variable and exposure variable are binary. Plasmode simulation samples subjects with replacement from the observed data, uses subjects' covariate data as is, and simulates exposure, outcome, or both.

Usage

```
PlasmodeBin(formulaOut = NULL, objectOut = NULL, formulaExp = NULL,
  objectExp = NULL, data, idVar, effectOR = 1, MMOut = 1, MMExp = 1,
  nsim, size, eventRate = NULL, exposedPrev = NULL)
```

Arguments

`formulaOut` An outcome model formula containing the binary outcome on the left-hand side and binary exposure along with potential confounders on the right-hand side.The functional form of the outcome model should be, Outcome ~ Exposure + Confounders. (Exposure main effect must be first independent variable)

`objectOut` A fitted model for the outcome model. The functional form of the fitted model for the outcome variable should be of form, Outcome ~ Exposure + Confounders.

`formulaExp` An exposure model formula containing the binary exposure on the left-hand side and potential confounders on the right-hand side. The functional form of the exposure model is, Exposure ~ Confounders.

`objectExp` A fitted model object for the exposure model.

`data` The dataset on which simulations are based. The data is required only when formulaOut or formulaExp or both are supplied to the argument.

`idVar` Name of the ID variable.

`effectOR` The desired treatment effect odds ratio. By default effectOR = 1.

`MMOut` A multiplier of confounder effects on outcome applied to the estimated log ORs in the outcome model. By default MMOut = 1 but one can specify a vector of length equivalent to the number of variables on the right-hand side of the outcome model.

`MMExp` A multiplier of confounder effects on exposure applied to the estimated log ORs in the exposure model. By default MMExp = 1 but one can specify a vector of length equivalent to the number of variables on the right-hand side of the exposure model.

`nsim` Number of desired simulated datasets.

`size` Desired size of simulated datasets (i.e., # of individuals).

`eventRate` Desired average event rate. Default is the event rate in the observed data.

`exposedPrev` Desired average exposure rate. Default is the esposure prevalence in the observed data.

Details

At least one of formulaOut, formulaExp, objectOut, and objectExp must be specified, and which of these are specified will determine what gets simulated and how. If objectOut or objectExp are specified, these objects are used as the base model for outcome and exposure simulation. If formulaOut or formulaExp are specified, then data should be given and base models are fit in the data using glm2 with the given formulas. If formulaOut or objectOut is specified, outcome will be simulated based on subjects' observed exposure. If formulaExp or objectExp is specified, exposure will be simulated. And if models are specified for both outcome and exposure, both variables will be simulated with simulated outcome dependent on the simulated exposure.

Value

PlasmodeBin returns true beta coefficients used to generate the outcome and the exposure. It also returns the relative risk and risk difference estimated by the plasmode simulated data along with the data frame with the simulated data, including sampled IDs for each of nsim datasets along with simulated outcomes, exposure, or both.

`TrueOutBeta` True beta coefficients used to generate the outcome.

`TrueExpBeta` True beta coefficients used to generate the exposure.

`RR` True relative risk estimated using the plasmode simulated data.

`RD` True risk difference estimated using the plasmode simulated data.

`Sim_Data` Plasmode simulated data, including sampled IDs for each of nsim datasets along with simulated outcomes, exposure, or both.

Author(s)

Jessica M. Franklin, Younathan Abdia, and Shirley Wang

Examples

```
{
library(mgcv)
library(nlme)
library(glm2)
library(arm)
library(MASS)
library(lme4)
library(epiDisplay)
library(foreign)
library(nnet)

data("Compaq")
levels(Compaq$stage) <- c(1,2,3,4)
Compaq$stage<-as.numeric(levels(Compaq$stage))[Compaq$stage]
## Creating the binary exposure variable
Compaq$exposure<-ifelse(Compaq$hospital == "Public hospital",1,0)
## Creating binary variables for some confounders
Compaq$ses1<-ifelse(Compaq$ses == "Poor",1,0)
Compaq$ses2<-ifelse(Compaq$ses == "Poor-middle",1,0)
```

```
Compaq$ses3<-ifelse(Compaq$ses == "High-middle",1,0)

Compaq$age1<-ifelse(Compaq$agegr == "<40",1,0)
Compaq$age2<-ifelse(Compaq$agegr == "40-49",1,0)
Compaq$age3<-ifelse(Compaq$agegr == "50-59",1,0)


## Creating the formulas for the outcome and the exposure model
form1<- status~ exposure + stage + ses1 + ses2 + ses3 + age1 + age2 + age3
form2<- exposure ~ stage + ses1 + ses2 + ses3 + age1 + age2 + age3

set.seed(111)
Bin_Form1<-PlasmodeBin(formulaOut=form1, objectOut=NULL,formulaExp=form2,
                     objectExp= NULL,data=Compaq,idVar="id",effectOR =1,
                     MMOut=c(1,1,2,1,1,2,1,2),MMExp=c(1,1,1,1,1,1,1),
                     nsim=2, size=nrow(Compaq), eventRate=NULL, exposedPrev=NULL)

Bin_Form2<-PlasmodeBin(formulaOut=form1, objectOut=NULL,formulaExp=NULL,
                      objectExp= NULL,data=Compaq,idVar="id",effectOR =1,
                      MMOut=c(1,1,2,1,1,2,1,2),MMExp=1, nsim=2,
                      size=nrow(Compaq), eventRate=NULL, exposedPrev=NULL)

Bin_Form3<-PlasmodeBin(formulaOut=NULL, objectOut=NULL,formulaExp=form2,
                      objectExp= NULL,data=Compaq,idVar="id",effectOR =1,
                      MMOut=1,MMExp=c(1,1,1,1,1,1,1), nsim=2,
                      size=nrow(Compaq), eventRate=NULL, exposedPrev=NULL)


###################################################################################################
## One can provide the fitted model for the outcome model and the exposure model estimated by
## glm, gam, and bayesglm. The functional form of the fitted model for the outcome variable should
## of the form Outcome ~ Exposure + Confounders. The functional form of the exposure model is,
## Exposure ~ Confounders.
####################################################################################################

Coeff1<- bayesglm(form1, family = "binomial", data=Compaq,control=glm.control(trace=TRUE))
Coeff2<- bayesglm(form2, family = "binomial", data=Compaq,control=glm.control(trace=TRUE))
sizesim<-nrow(model.matrix(Coeff1))
sizesim1<-nrow(model.matrix(Coeff2))

Bin_Obj1<-PlasmodeBin(formulaOut=NULL, objectOut=Coeff1,formulaExp=NULL,
                     objectExp = Coeff2, idVar=Compaq$id,effectOR =1,
                     MMOut=c(1.5,1,2,1,1,1,1,1),MMExp=c(1,1,1,1,1,1,1),
                     nsim=2, size=sizesim, eventRate=NULL, exposedPrev=NULL)

Bin_Obj2<-PlasmodeBin(formulaOut=NULL, objectOut=Coeff1,formulaExp=NULL,
                     objectExp = NULL,idVar=Compaq$id,effectOR =1,
                     MMOut=c(1.5,1,2,1,1,1,1,1),MMExp=1,
                     nsim=2, size=sizesim, eventRate=NULL, exposedPrev=NULL)

Bin_Obj3<-PlasmodeBin(formulaOut=NULL, objectOut=NULL,formulaExp=NULL,
                     objectExp = Coeff2,idVar=Compaq$id,effectOR =1, MMOut=1,
                     MMExp=c(1,1,1,1,1,1,1),
                   nsim=2, size=sizesim1, eventRate=NULL, exposedPrev=NULL)
}
```

PlasmodeCont *Performs the plasmode simulation*

Description

Creates 'plasmode' simulated datasets based on a given dataset when the outcome variable is continuous and exposure variable is binary. Plasmode simulation samples subjects with replacement from the observed data, uses subjects' covariate data as is, and simulates exposure, outcome, or both.

Usage

```
PlasmodeCont(formulaOut = NULL, objectOut = NULL, formulaExp = NULL,
  objectExp = NULL, data, idVar, effectOR = 1, MMOut = 1, MMExp = 1,
  nsim, size, eventRate = NULL, exposedPrev = NULL)
```

Arguments

formulaOut	An outcome model formula containing the continuous outcome on the left-hand side and binary exposure along with potential confounders on the right-hand side.The functional form of the outcome model should be, Outcome ~ Exposure + Confounders. (Exposure main effect must be first independent variable).
objectOut	A fitted model for the outcome model. The functional form of the fitted model for the outcome variable should be of form, Outcome ~ Exposure + Confounders.
formulaExp	An exposure model formula containing the binary exposure on the left-hand side and potential confounders on the right-hand side. The functional form of the exposure model is, Exposure ~ Confounders.
objectExp	A fitted model object for the exposure model.
data	The dataset on which simulations are based. The data is required only when formulaOut or formulaExp or both are supplied to the argument.
idVar	Name of the ID variable
effectOR	The desired treatment effect odd ratio. By default effectOR = 1.
MMOut	A multiplier of confounder effects on outcome applied to the estimated log ORs in the outcome model. By default MMOut = 1 but one can specify a vector of length equivalent to the number of variables on the right-hand side of the outcome model.
MMExp	A multiplier of confounder effects on exposure applied to the estimated log ORs in the exposure model. By default MMExp = 1 but one can specify a vector of length equivalent to the number of variables on the right-hand side of the exposure model.
nsim	Number of desired simulated datasets.

`size` Desired size of simulated datasets (i.e., # of individuals).

`eventRate` Desired average event rate. Default is the event rate in the observed data.

`exposedPrev` Desired average exposure rate. Default is the esposure prevalence in the observed data.

Details

At least one of formulaOut, formulaExp, objectOut, and objectExp must be specified, and which of these are specified will determine what gets simulated and how. If objectOut or objectExp are specified, these objects are used as the base model for outcome and exposure simulation. If formulaOut or formulaExp are specified, then data should be given and base models are fit in the data using glm2 with the given formulas. If formulaOut or objectOut is specified, outcome will be simulated based on subjects' observed exposure. If formulaExp or objectExp is specified, exposure will be simulated. And if models are specified for both outcome and exposure, both variables will be simulated with simulated outcome dependent on the simulated exposure.

Value

PlasmodeCont returns true beta coefficients used to generate the outcome and the exposure. It also returns the relative risk and risk difference estimated by the plasmode simulated data along with the data frame with the simulated data, including sampled IDs for each of nsim datasets along with simulated outcomes, exposure, or both.

`TrueOutBeta` True beta coefficients used to generate the outcome.

`TrueExpBeta` True beta coefficients used to generate the exposure.

`RR` True relative risk estimated using the plasmode simulated data.

`RD` True risk difference estimated using the plasmode simulated data.

`Sim_Data` Plasmode simulated data, including sampled IDs for each of nsim datasets along with simulated outcomes, exposure, or both.

Author(s)

Jessica M. Franklin, Younathan Abdia, and Shirley Wang

Examples

```
{
## Example for using the PlasmodeCont
library(twang)
library(gbm)
library(lattice)
library(parallel)
library(survey)
library(grid)
library(Matrix)
library(xtable)
library(latticeExtra)
library(RColorBrewer)
library(arm)
set.seed(1)
data("lalonde")
## Creating the ID variable
lalonde$id <- 1:nrow(lalonde)

str(lalonde)
## Example for PlasmodeCont when the outcome and exposure models formulas are provided.
form1<- re78 ~ treat + age + educ + black + hisp+ nodegr  + married + re74 + re75
form2<- treat ~ age + educ + black + hisp + nodegr + married + re74 + re75
Cont_Form1<-PlasmodeCont(formulaOut=form1, objectOut = NULL,formulaExp=form2,objectExp = NULL,
                        data=lalonde,idVar="id",effectOR =0, MMOut=c(0,1,2,1,1,1,2,2,1),
                        MMExp=c(1,2,1,1,1,2,2,1),nsim=2, size=nrow(lalonde),
                        eventRate=NULL, exposedPrev=NULL)
Cont_Form2<-PlasmodeCont(formulaOut=form1, objectOut = NULL,formulaExp=NULL,objectExp = NULL,
                   data=lalonde,idVar="id",effectOR =0, MMOut=c(0,1,2,1,1,1,2,2,1),MMExp=1,
                        nsim=2, size=nrow(lalonde), eventRate=NULL, exposedPrev=NULL)
Cont_Form3<-PlasmodeCont(formulaOut=NULL, objectOut = NULL,formulaExp=form2,objectExp = NULL,
                   data=lalonde,idVar="id",effectOR =0, MMOut=1,MMExp=c(1,2,1,1,1,2,2,1),
                        nsim=2, size=nrow(lalonde), eventRate=NULL, exposedPrev=NULL)
## Example for PlasmodeCont when the fitted model objects are provided.
####################################################################################################
## One can provide the fitted model for the outcome model and the exposure model estimated by
## glm, gam, and bayesglm. The functional form of the fitted model for the outcome variable should
## of the form Outcome ~ Exposure + Confounders. The functional form of the exposure model is,
## Exposure ~ Confounders.
#####################################################################################################
Coeff1c<- bayesglm(form1, family = "gaussian", data=lalonde,control=glm.control(trace=TRUE))
Coeff2c<- bayesglm(form2, family = "binomial", data=lalonde,control=glm.control(trace=TRUE))

sizesim<-nrow(model.matrix(Coeff1c))
sizesim1<-nrow(model.matrix(Coeff2c))

Cont_Obj1<-PlasmodeCont(formulaOut=NULL, objectOut = Coeff1c,formulaExp=NULL,objectExp = Coeff2c,
                       idVar=lalonde$id,effectOR =0, MMOut=c(0,1,2,1,1,1,2,2,1),
                       MMExp=c(1,2,1,1,1,2,2,1),
                       nsim=2, size=nrow(lalonde), eventRate=NULL, exposedPrev=NULL)

Cont_Obj2<-PlasmodeCont(formulaOut=NULL, objectOut = Coeff1c,formulaExp=NULL,objectExp = NULL,
                       idVar=lalonde$id,effectOR =1, MMOut=c(0,1,2,1,1,1,2,2,1),MMExp=1,
                       nsim=2, size=nrow(lalonde), eventRate=NULL, exposedPrev=NULL)

Cont_Obj3<-PlasmodeCont(formulaOut=NULL, objectOut = NULL,formulaExp=NULL,objectExp = Coeff2c,
                       idVar=lalonde$id,effectOR =1, MMOut=c(0,1,2,1,1,1,2,2,1),MMExp=1,
                       nsim=2, size=nrow(lalonde), eventRate=NULL, exposedPrev=NULL)
}
```

PlasmodeSur *Performs the plasmode simulation*

Description

Creates 'plasmode' simulated datasets based on a given dataset when the outcome variable is time to event and exposure variable are binary. Plasmode simulation samples subjects with replacement from the observed data, uses subjects' covariate data as is, and simulates exposure, outcome, or both.

Usage

```
PlasmodeSur(formulaOut = NULL, formulaCen = NULL, objectOut = NULL,
  objectCen = NULL, formulaExp = NULL, objectExp = NULL, data, idVar,
  effectOR = 1, MMOut = 1, MMExp = 1, nsim, size, eventRate = NULL,
  exposedPrev = NULL)
```

Arguments

`formulaOut` An outcome model formula for estimating the hazard of outcome event.The functional form of the outcome model should be, Surv(data$time, data$event)~ Exposure + Confounders, where data is the dataset on which simulations are based, time is the follow-up time for the right-censored data and event is the status indicator. Exposure main effect must be first independent variable.

`formulaCen` An outcome model formula for estimating the hazard of censoring.The functional form of the outcome model should be, Surv(data$time, !data$event)~ Exposure + Confounders, where data is the dataset on which simulations are based, time is the follow-up time for the right-censored data and event is the status indicator. Exposure main effect must be first independent variable.

`objectOut` A fitted model object for the hazard of outcome.The functional form of the fitted model object should be of form coxph(Surv(data$time, data$event)~ Exposure + Confounders, data,x=TRUE), where coxph fits the Cox proportional hazard model, data is the dataset on which simulations are based, time is the follow-up time for the right-censored data and event is the status indicator. Exposure main effect must be first independent variable.

`objectCen` A fitted model object for the hazard of censoring.The functional form of the fitted model object should be of form coxph(Surv(data$time, !data$event)~ Exposure + Confounders, data,x=TRUE), where coxph fits the Cox proportional hazard model, data is the dataset on which simulations are based, time is the follow-up time for the right-censored data and event is the status indicator. Exposure main effect must be first independent variable.

`formulaExp` An exposure model formula containing the binary exposure on the left-hand side and potential confounders on the right-hand side. The functional form of the exposure model is, Exposure ~ Confounders.

`objectExp` A fitted model object for the exposure model.

`data` The dataset on which simulations are based.The data is required only when formulaOut, formulaCen or formulaExp or both are supplied to the argument.

`idVar` Name of the ID variable

`effectOR` The desired treatment effect odds ratio. By default effectOR = 1.

`MMOut` Multiplier of confounder effects on outcome on the log-scale. By default MMOut = 1 but one can specify a vector of length equivalent to the number of variables on the right-hand side of the outcome model.

`MMExp` Multiplier of confounder effects on exposure. By default MMExp = 1 but one can specify a vector of length equivalent to the number of variables on the right-hand side of the exposure model.

`nsim` Number of desired simulated datasets.

`size` Desired size of simulated datasets (i.e., # of individuals).

`eventRate` Desired average event rate. Default is the event rate in the observed data.

`exposedPrev` Desired average exposure rate. Default is the exposure prevalence in the observed data.

Details

At least one of formulaOut, formulaCen, formulaExp, objectOut,objectCen, and objectExp must be specified, and which of these are specified will determine what gets simulated and how. If objectOut and objectCen or objectExp are specified, these objects are used as the base model for outcome and exposure simulation. If formulaOut and formulaCen or formulaExp are specified, then data should be given and base models are fit in the data using coxph with the given formulas. If formulaOut and formulaCen or objectOut and objectCen is specified, outcome will be simulated based on subjects' observed exposure. If formulaExp or objectExp is specified, exposure will be simulated. And if models are specified for both outcome and exposure, both variables will be simulated with simulated outcome dependent on the simulated exposure.

Value

PlasmodeSur returns true beta coefficients used to generate the outcome and the exposure. PlasmodeSur also returns the data frame with the simulated data, including sampled IDs for each of nsim datasets along with simulated outcomes, exposure, or both.

`TrueOutBeta` True beta coefficients used to generate the outcome.

`TrueExpBeta` True beta coefficients used to generate the exposure.

`Sim_Data` Plasmode simulated data, including sampled IDs for each of nsim datasets along with simulated outcomes, exposure, or both.

Author(s)

Jessica M. Franklin, Younathan Abdia, and Shirley Wang

Examples

```
{
library(survival)
library(splines)
library(glm2)
## Creating data set for simulation
lung <- lung[complete.cases(lung),]
lung$id <- 1:nrow(lung)
lung$meal.cal <- ifelse(lung$meal.cal > 1000, 1, 0)
lung$status <- lung$status - 1

## Formulas for estimating the hazard of outcome event, the hazard of censoring and exposure.

form1<-Surv(lung$time, lung$status)~meal.cal+age+sex+ph.ecog+ph.karno
form2<-Surv(lung$time, !lung$status)~meal.cal+age+sex+ph.ecog+ph.karno
form3<- meal.cal~age+sex+ph.ecog+ph.karno

Sur_Form1<-PlasmodeSur(formulaOut=form1,formulaCen=form2, objectOut=NULL, objectCen = NULL,
          formulaExp=form3,objectExp=NULL,data=lung,idVar="id",effectOR =1, MMOut=c(0.5,2,2,1,3),
            MMExp=c(2,2,2,2), nsim=3, size=nrow(lung), eventRate=NULL, exposedPrev=NULL)

Sur_Form2<-PlasmodeSur(formulaOut=form1,formulaCen=form2, objectOut=NULL, objectCen = NULL,
          formulaExp=NULL,objectExp=NULL,data=lung,idVar="id",effectOR =1, MMOut=c(1,2,2,1,3),
            MMExp=c(1,1,1,1),nsim=3, size=nrow(lung), eventRate=NULL, exposedPrev=NULL)

Sur_Form3<-PlasmodeSur(formulaOut=NULL,formulaCen=NULL, objectOut=NULL, objectCen = NULL,
          formulaExp=form3,objectExp=NULL,data=lung,idVar="id",effectOR =1, MMOut=c(1,2,2,1,3),
            MMExp=c(1,1,1,1),nsim=3, size=nrow(lung), eventRate=NULL, exposedPrev=NULL)

## Objects for the hazard of the outcome event, hazard for censoring and the exposure.

smod1 <- coxph(Surv(lung$time, lung$status)~meal.cal+age+sex+ph.ecog+ph.karno, data = lung,x=TRUE)
smod2 <- coxph(Surv(lung$time, !lung$status)~meal.cal+age+sex+ph.ecog+ph.karno, data = lung,x=TRUE)
pmod1<-glm2(meal.cal~age+sex+ph.ecog+ph.karno, data = lung,family = "binomial",
            control=glm.control(trace=TRUE))

Sur_Obj1<-PlasmodeSur(formulaOut=NULL,formulaCen=NULL, objectOut=smod1,objectCen = smod2,
          formulaExp=NULL,objectExp=pmod1,idVar=lung$id, effectOR =1, MMOut=c(1,2,2,1,3),
            MMExp=1, nsim=3,size=nrow(lung), eventRate=0.5, exposedPrev=NULL)

Sur_Obj2<-PlasmodeSur(formulaOut=NULL,formulaCen=NULL, objectOut=smod1,objectCen = smod2,
          formulaExp=NULL,objectExp=NULL,idVar=lung$id, effectOR =1.5, MMOut=c(1,2,2,1,3),
            MMExp=1, nsim=3,size=nrow(lung), eventRate=0.5, exposedPrev=NULL)

Sur_Obj3<-PlasmodeSur(formulaOut=NULL,formulaCen=NULL, objectOut=NULL,objectCen = NULL,
          formulaExp=NULL,objectExp=pmod1,idVar=lung$id,effectOR =1, MMOut=c(1,2,2,1,3),
            MMExp=1, nsim=3,size=nrow(lung), eventRate=0.5, exposedPrev=NULL)
}
```

Index