Package ‘interactionRCS’

March 1, 2022

Type Package

Title Calculate Estimates in Models with Interaction

Version 0.1.0

Date 2022-02-25

Description A tool to calculate and plot estimates from models
in which an interaction between the main predictor and a continuous covariate has been specified.
Methods used in the package refer to Harrell Jr FE (2015, ISBN:9783319330396);
Durrleman S, Simon R. (1989) <doi:10.1002/sim.4780080504>; Greenland S. (1995) <doi:10.1097/00001648-199507000-00005>.

License GPL-3

Depends R (>= 3.6)

Imports graphics, grDevices, stats, utils, methods, parallel,
survival, msm, rms, boot, pspline, pryr, mlbench

Suggests knitr, rmarkdown, dplyr

VignetteBuilder knitr

RoxygenNote 7.1.1

Encoding UTF-8

LazyData true

NeedsCompilation no

Author Giorgio Melloni [aut, cre] (<https://orcid.org/0000-0001-6371-1334>),
Andrea Bellavia [aut] (<https://orcid.org/0000-0003-4988-4532>)

Maintainer Giorgio Melloni <melloni.giorgio@gmail.com>

Repository CRAN

Date/Publication 2022-03-01 08:50:02 UTC

R topics documented:

intEST *Returns the estimates of for an unspecified interaction model*

Description

This function is a dispatcher that generate OR, HR or linear estimates values for a simple or restricted cubic spline interaction model from a logistic, Cox or linear regression

Usage

```
intEST(
  var2values,
  model,
  data,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values`	numeric vector of var2 points to estimate
`model`	model of class cph, coxph, lrm, glm or Glm. If data is NULL, the function expects to find the data in model$x.
`data`	data used in the model. If absent, we will attempt to recover the data from the model. Only used for bootstrap and glm class models
`var1`	variable that increases by 1 unit from 0
`var2`	variable to spline. var2values belong to var2
`ci`	calculate 95% CI?
`conf`	confidence level. Default 0.95

`ci.method` confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)

`ci.boot.method` one of the available bootstrap CI methods from `boot.ci`. Default percentile

`R` number of bootstrap samples if ci.method = "bootstrap". Default 100

`parallel` can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore

`...` other parameters for boot

Value

if ci = FALSE, a dataframe with initial values and OR/HR/linear estimates , if ci = TRUE a dataframe with 5 columns, initial values, OR/HR/linear estimates, lower CI, upper CI and SE

Examples

```
library(rms)
library(mlbench)
data(PimaIndiansDiabetes)
# Set age on a 5-year scale
PimaIndiansDiabetes$age <- PimaIndiansDiabetes$age/5
# Recode diabetes as 0/1
PimaIndiansDiabetes$diabetes <- ifelse(PimaIndiansDiabetes$diabetes==”pos” , 1 , 0)
# Logistic model predicting diabetes over BMI, age and glucose
myformula <- diabetes ~ mass + age * rcs( glucose , 3 )
model <- lrm(myformula , data = PimaIndiansDiabetes )
intEST( var2values = 20:80
       , model = model , data = PimaIndiansDiabetes , var1 =”age”, var2=”glucose”
       , ci=TRUE , conf = 0.95 , ci.method = ”delta”)
# Linear model predicting BMI over diabetes, age and glucose
myformula2 <- mass ~ diabetes + age * rcs( glucose , 3 )
model2 <- glm(myformula2 , data = PimaIndiansDiabetes , family = ”gaussian”)
intEST( var2values = 20:80
       , model = model2 , data = PimaIndiansDiabetes , var1 =”age”, var2=”glucose”
       , ci=TRUE , conf = 0.95 , ci.method = ”delta”)
```

`linLIN` *Linear regression interaction estimates*

Description

Generate linear estimates for a 1 unit increase in a variable at specified points of another interacting variable in a linear interaction model

Usage

```
linLIN(
  var2values,
  model,
  data,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values` numeric vector of var2 points to estimate

`model` model of class rms::Glm or stats::glm family gaussian. If data is NULL, the function expects to find the data in model$x

`data` data used in the model. If absent, it will attempt to recover the data from the model object. Only used for bootstrap CI

`var1` variable that increases by 1 unit from 0

`var2` variable to spline. var2values belong to var2

`ci` calculate 95% CI?

`conf` confidence level. Default 0.95

`ci.method` confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)

`ci.boot.method` one of the available bootstrap CI methods from `boot.ci`. Default percentile

`R` number of bootstrap samples if ci.method = "bootstrap". Default 100

`parallel` can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore

`...` other parameters for boot

Value

if ci = FALSE, a vector of estimate of length(var2values), if ci = TRUE a dataframe with 5 columns, initial values, linear estimates, lower CI, upper CI and SE

Examples

```
library(rms)
library(mlbench)
data(PimaIndiansDiabetes)
```

```
# Recode diabetes as 0/1
PimaIndiansDiabetes$diabetes <- ifelse(PimaIndiansDiabetes$diabetes=="pos" , 1 , 0)
myformula <- glucose ~ mass + diabetes * age
model <- glm(myformula , data = PimaIndiansDiabetes ,family=gaussian)
# Show the effect on glucose of being diabetic at age 20 to 80
linLIN( var2values = 20:80
       , model = model , data = PimaIndiansDiabetes , var1 ="diabetes", var2="age"
       , ci=TRUE , conf = 0.95 , ci.method = "delta")
```

---

loglinHR *Linear interaction HR*

Description

Generate HR values for a 1 unit increase in a variable at specified points of another interacting variable in a simple Cox interaction model

Usage

```
loglinHR(
  var2values,
  model,
  data,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values`	numeric vector of var2 points to estimate
`model`	model of class coxph or cph. If data is NULL, the function expects to find the data in model$x
`data`	data used in the model. If absent, it will attempt to recover the data from the model object. Only used for bootstrap CI
`var1`	variable that increases by 1 unit from 0
`var2`	variable to spline. var2values belong to var2
`ci`	calculate 95% CI?
`conf`	confidence level. Default 0.95

`ci.method` confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)

`ci.boot.method` one of the available bootstrap CI methods from `boot.ci`. Default percentile

`R` number of bootstrap samples if ci.method = "bootstrap". Default 100

`parallel` can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore

`...` other parameters for boot

Value

if ci = FALSE, a vector of estimate of length(var2values), if ci = TRUE a dataframe with 5 columns, initial values, HR, lower CI, upper CI and SE

Examples

```
library(survival)
data(cancer)
myformula <- Surv(time, status) ~ ph.karno + ph.ecog + age*sex
model <- coxph(myformula , data = lung )
loglinHR( var2values = 40:80
                    , model = model , data = lung , var1 ="sex", var2="age"
                    , ci=TRUE , conf = 0.95 , ci.method = "delta")
```

loglinOR *Linear interaction OR*

Description

Generate OR values for a 1 unit increase in a variable at specified points of another interacting variable in a simple logistic interaction model

Usage

```
loglinOR(
  var2values,
  model,
  data,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values` numeric vector of var2 points to estimate

`model` model of class lrm Glm or glm. If data is NULL, the function expects to find the data in model$x

`data` data used in the model. If absent, it will attempt to recover the data from the model object. Only used for bootstrap CI

`var1` variable that increases by 1 unit from 0

`var2` variable to spline. var2values belong to var2

`ci` calculate 95% CI?

`conf` confidence level. Default 0.95

`ci.method` confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)

`ci.boot.method` one of the available bootstrap CI methods from `boot.ci`. Default percentile

`R` number of bootstrap samples if ci.method = "bootstrap". Default 100

`parallel` can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore

`...` other parameters for boot

Value

if ci = FALSE, a vector of estimate of length(var2values), if ci = TRUE a dataframe with 5 columns, initial values, OR, lower CI, upper CI and SE

Examples

```
library(rms)
library(mlbench)
data(PimaIndiansDiabetes)
# Set age on a 5-year scale
PimaIndiansDiabetes$age <- PimaIndiansDiabetes$age/5
# Recode diabetes as 0/1
PimaIndiansDiabetes$diabetes <- ifelse(PimaIndiansDiabetes$diabetes=="pos" , 1 , 0)
myformula <- diabetes ~ mass + age * glucose
model <- glm(myformula , data = PimaIndiansDiabetes , family = binomial())
loglinOR( var2values = 20:80
       , model = model , data = PimaIndiansDiabetes , var1 ="age", var2="glucose"
       , ci=TRUE , conf = 0.95 , ci.method = "delta")
```

plotINT *Plot the result of HR, OR or linear estimates*

Description

Create a spline var2 by 1 unit increase of var1

Usage

```
plotINT(
  x,
  xlab = "",
  main = "",
  log = FALSE,
  ylab = NULL,
  line1 = TRUE,
  color = "dodgerblue"
)
```

Arguments

x	data.frame calculated using any of the function of this package
xlab	xlab name
main	plot title
log	if TRUE, plot the estimate in log scale
ylab	ylab name. Default is the estimate column name if log=FALSE otherwise Estimate(log scale)
line1	if TRUE, plot horizontal line on 1 or 0 (if log=TRUE)
color	line color. Default dodgerblue

Value

simple splined plot of estimates of var1 at var2 values

Examples

```
library(rms)
library(survival)
data(cancer)
myformula <- Surv(time, status) ~ ph.karno + ph.ecog + rcs(age, 3)*sex
model <- cph(myformula , data = lung )
myHR <- rcsHR( var2values = 40:80
               , model = model , data = lung , var1 ="sex", var2="age"
               , ci=TRUE , conf = 0.95 , ci.method = "delta")
plotINT(myHR , ylab = "HR of male VS female" , xlab = "Age")
```

rcsHR *Restricted cubic spline interaction HR*

Description

Generate HR values in a Cox model for a 1 unit increase in a variable at specified points of another interacting variable splined with rcs(df = 3)

Usage

```
rcsHR(
  var2values,
  model,
  data = NULL,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values`	numeric vector of var2 points to estimate
`model`	model of class cph or coxph. If data is NULL, the function expects to find the data in model$x.
`data`	data used in the model. If absent, we will attempt to recover the data from the model. Only used for bootstrap and coxph models
`var1`	variable that increases by 1 unit from 0.
`var2`	variable to spline. var2values belong to var2
`ci`	calculate 95% CI?
`conf`	confidence level. Default 0.95
`ci.method`	confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)
`ci.boot.method`	one of the available bootstrap CI methods from `boot.ci`. Default percentile
`R`	number of bootstrap samples if ci.method = "bootstrap". Default 100
`parallel`	can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore
`...`	other parameters for boot

Value

if ci = FALSE, a dataframe with initial values and HR , if ci = TRUE a dataframe with 5 columns, initial values, HR, lower CI, upper CI and SE

Examples

```
library(survival)
library(rms)
data(cancer)
myformula <- Surv(time, status) ~ ph.karno + ph.ecog + rcs(age,3)*sex
model <- cph(myformula , data = lung )
rcsHR( var2values = 40:80
       , model = model , data = lung , var1 ="sex", var2="age"
       , ci=TRUE , conf = 0.95 , ci.method = "delta")
```

---

rcsLIN *Restricted cubic spline interaction linear estimates*

Description

Generate estimates in a linear model for a 1 unit increase in a variable at specified points of another interacting variable splined with rcs(df = 3)

Usage

```
rcsLIN(
  var2values,
  model,
  data = NULL,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values` numeric vector of var2 points to estimate

`model` model of class rms::Glm or stats::glm family gaussian. If data is NULL, the function expects to find the data in model$x.

`data` data used in the model. If absent, we will attempt to recover the data from the model. Only used for bootstrap

`var1`	variable that increases by 1 unit from 0
`var2`	variable to spline. var2values belong to var2
`ci`	calculate 95% CI?
`conf`	confidence level. Default 0.95
`ci.method`	confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)
`ci.boot.method`	one of the available bootstrap CI methods from `boot.ci`. Default percentile
`R`	number of bootstrap samples if ci.method = "bootstrap". Default 100
`parallel`	can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore
`...`	other parameters for boot

Value

if ci = FALSE, a dataframe with initial values and linear estimates , if ci = TRUE a dataframe with 5 columns, initial values, linear estimates, lower CI, upper CI and SE

Examples

```
library(rms)
library(mlbench)
data(PimaIndiansDiabetes)
# Recode diabetes as 0/1
PimaIndiansDiabetes$diabetes <- ifelse(PimaIndiansDiabetes$diabetes==”pos” , 1 , 0)
myformula <- glucose ~ mass + diabetes * rcs(age, 3)
model <- glm(myformula , data = PimaIndiansDiabetes , family=”gaussian”)
# Show the effect on glucose of being diabetic at age 20 to 80
rcsLIN( var2values = 20:80
       , model = model , data = PimaIndiansDiabetes , var1 =”diabetes”, var2=”age”
       , ci=TRUE , conf = 0.95 , ci.method = ”delta”)
```

`rcsOR` *Restricted cubic spline interaction OR*

Description

Generate OR values in a logistic model for a 1 unit increase in a variable at specified points of another interacting variable splined with rcs(df = 3)

Usage

```
rcsOR(
  var2values,
  model,
  data = NULL,
  var1,
  var2,
  ci = TRUE,
  conf = 0.95,
  ci.method = "delta",
  ci.boot.method = "perc",
  R = 100,
  parallel = "multicore",
  ...
)
```

Arguments

`var2values` numeric vector of var2 points to estimate

`model` model of class lrm, Glm or glm family binomial. If data is NULL, the function expects to find the data in model$x.

`data` data used in the model. If absent, we will attempt to recover the data from the model. Only used for bootstrap and glm class models

`var1` variable that increases by 1 unit from 0

`var2` variable to spline. var2values belong to var2

`ci` calculate 95% CI?

`conf` confidence level. Default 0.95

`ci.method` confidence interval method. "delta" performs delta method. "bootstrap" performs bootstrapped CI (slower)

`ci.boot.method` one of the available bootstrap CI methods from `boot.ci`. Default percentile

`R` number of bootstrap samples if ci.method = "bootstrap". Default 100

`parallel` can take values "no", "multicore", "snow" if ci.method = "bootstrap". Default multicore

`...` other parameters for boot

Value

if ci = FALSE, a dataframe with initial values and OR , if ci = TRUE a dataframe with 5 columns, initial values, OR, lower CI, upper CI and SE

Examples

```
library(rms)
library(mlbench)
data(PimaIndiansDiabetes)
# Set age on a 5-year scale
PimaIndiansDiabetes$age <- PimaIndiansDiabetes$age/5
# Recode diabetes as 0/1
PimaIndiansDiabetes$diabetes <- ifelse(PimaIndiansDiabetes$diabetes=="pos" , 1 , 0)
myformula <- diabetes ~ mass + age * rcs( glucose , 3 )
model <- lrm(myformula , data = PimaIndiansDiabetes )
rcsOR( var2values = 20:80
      , model = model , data = PimaIndiansDiabetes , var1 ="age", var2="glucose"
      , ci=TRUE , conf = 0.95 , ci.method = "delta")
```

umaru *UMARU IMPACT Study data*

Description

A subset of data from the University of Massachusets Aids Research Unit (UMARU) IMPACT study.

Usage

```
umaru
```

Format

A data frame with 575 rows and 10 variables

X observation count

id identification code

age Age in years

beckscore Beck Depression Score at admission

ndrugtx Number of prior drug treatments

treat treat

site site

los los

time time to event

censor censor event

heroin heroin use at admission

cocaine cocaine use at admission

drugtx Prior Drug treatment, yes/no

ivdrug Prior IV drug treatment

nonwhite 1 if non white, 0 otherwise

Source

ftp://ftp.wiley.com/public/sci_tech_med/logistic

Index