Package 'riskRegression'

October 5, 2015

Type Package

Title Risk Regression Models for Survival Analysis with Competing Risks

Version 1.1.7

Author Thomas Alexander Gerds, Thomas Harder Scheike

Depends R (>= 2.9.0), prodlim (>= 1.4.9)

Imports rms (>= 4.2-0), pec (>= 2.4.3), cmprsk (>= 2.2-7), survival (>= 2.37-7), randomForestSRC (>= 1.5.5)

Suggests lava

Maintainer Thomas Alexander Gerds <tag@biostat.ku.dk>

Description Risk regression models for survival analysis with and without competing risks.

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-10-05 12:58:22

R topics documented:

CSC *Cause-specific Cox proportional hazard regression*

Description

Interface for fitting cause-specific Cox proportional hazard regression models in competing risk.

Usage

```
CSC(formula, data, cause, survtype = "hazard", fitter = "coxph", ...)
```

Arguments

formula	A list of formulae, one for each cause, each specifying a cause-specific Cox regression model.
data	A data in which to fit the models.
cause	The cause of interest. Defaults to the first cause.
survtype	Either "hazard" (the default) or "survival". If "hazard" fit cause-specific Cox regression models for all causes. If "survival" fit one cause-specific Cox regression model for the cause of interest and also a Cox regression model for event-free survival.
fitter	Routine to fit the Cox regression models. If coxph use survival::coxph else use rms::cph. The latter is much faster when it comes to prediction.
...	Arguments given to coxph.

Value

models	a list with the fitted (cause-specific) Cox regression objects
response	the event history response
eventTimes	the sorted (unique) event times
survtype	the value of survtype
theCause	the cause of interest. see cause
causes	the other causes

Author(s)

Thomas A. Gerds <tag@biostat.ku.dk>

Ulla B. Mogensen <ulmo@biostat.ku.dk>

See Also

coxph

Examples

```
library(riskRegression)
library(prodlim)
library(pec)
library(survival)
data(Melanoma)
## fit two cause-specific Cox models
## different formula for the two causes
fit1 <- CSC(list(Hist(time,status)~sex,Hist(time,status)~invasion+epicel+age),
            data=Melanoma)
print(fit1)
fit1a <- CSC(list(Hist(time,status)~sex+age,Hist(time,status)~invasion+epicel+log(thick)),
             data=Melanoma,
             survtype="surv")
print(fit1a)

## same formula for both causes
fit2 <- CSC(Hist(time,status)~invasion+epicel+age,
            data=Melanoma)
print(fit2)

## combine a cause-specific Cox regression model for cause 2
## and a Cox regression model for the event-free survival:
## different formula for cause 2 and event-free survival
fit3 <- CSC(list(Hist(time,status)~sex+invasion+epicel+age,
                 Hist(time,status)~invasion+epicel+age),
            survtype="surv",
            data=Melanoma)
print(fit3)

## same formula for both causes
fit4 <- CSC(Hist(time,status)~invasion+epicel+age,
            data=Melanoma,
            survtype="surv")
print(fit4)

## strata
fit5 <- CSC(Hist(time,status)~invasion+epicel+age+strata(sex),
            data=Melanoma,
            survtype="surv")
print(fit5)

## sanity check

cox1 <- coxph(Surv(time,status==1)~invasion+epicel+age+strata(sex),data=Melanoma)
cox2 <- coxph(Surv(time,status!=0)~invasion+epicel+age+strata(sex),data=Melanoma)
all.equal(cox1,fit5$models[[1]])
all.equal(cox2,fit5$models[[2]])

## predictions
##
## survtype = "hazard": predictions for both causes can be extracted
```

```
## from the same fit
library(pec)
fit2 <- CSC(Hist(time,status)~invasion+epicel+age, data=Melanoma)
pec:::predictEventProb(fit2,cause=1,newdata=Melanoma[c(17,99,108),],times=c(100,1000))
pec:::predictEventProb(fit2,cause=2,newdata=Melanoma[c(17,99,108),],times=c(100,1000))

## survtype = "surv" we need to change the cause of interest

fit5.2 <- CSC(Hist(time,status)~invasion+epicel+age+strata(sex),
            data=Melanoma,
            survtype="surv",cause=2)
## now this does not work
try(pec:::predictEventProb(fit5.2,cause=1,newdata=Melanoma,times=4))

## but this does
try(pec:::predictEventProb(fit5.2,cause=2,newdata=Melanoma,times=100))
```

FGR *Formula wrapper for crr from cmprsk*

Description

Formula interface for Fine-Gray regression competing risk models.

Usage

```
FGR(formula, data, cause = 1, y = TRUE, ...)
```

Arguments

formula	A formula whose left hand side is a Hist object – see Hist. The right hand side specifies (a linear combination of) the covariates. See examples below.
data	A data.frame in which all the variables of formula can be interpreted.
cause	The failure type of interest. Defaults to 1.
y	logical value: if TRUE, the response vector is returned in component response.
...	...

Details

Formula interface for the function crr from the cmprsk package.

The function crr allows to multiply some covariates by time before they enter the linear predictor. This can be achieved with the formula interface, however, the code becomes a little cumbersome. See the examples.

Value

See crr.

Author(s)

Thomas Alexander Gerds <tag@biostat.ku.dk>

References

Gerds, TA and Scheike, T and Andersen, PK (2011) Absolute risk regression for competing risks: interpretation, link functions and prediction Research report 11/7. Department of Biostatistics, University of Copenhagen

See Also

riskRegression

Examples

```
library(prodlim)
library(survival)
library(pec)
library(cmprsk)
library(lava)
d <- SimCompRisk(100)
f1 <- FGR(Hist(time,cause)~X1+X2,data=d)
print(f1)

## crr allows that some covariates are multiplied by
## a function of time (see argument tf of crr)
## by FGR uses the identity matrix
f2 <- FGR(Hist(time,cause)~cov2(X1)+X2,data=d)
print(f2)

## same thing, but more explicit:
f3 <- FGR(Hist(time,cause)~cov2(X1)+cov1(X2),data=d)
print(f3)

## both variables can enter cov2:
f4 <- FGR(Hist(time,cause)~cov2(X1)+cov2(X2),data=d)
print(f4)

## change the function of time
qFun <- function(x){x^2}
noFun <- function(x){x}
sqFun <- function(x){x^0.5}

## multiply X1 by time^2 and X2 by time:
f5 <- FGR(Hist(time,cause)~cov2(X1,tf=qFun)+cov2(X2),data=d)
print(f5)
print(f5$crrFit)
## same results as crr
with(d,crr(ftime=time,
           fstatus=cause,
           cov2=d[,c("X1","X2")],
           tf=function(time){cbind(qFun(time),time)}))
```

```
## still same result, but more explicit
f5a <- FGR(Hist(time,cause)~cov2(X1,tf=qFun)+cov2(X2,tf=noFun),data=d)
f5a$crrFit

## multiply X1 by time^2 and X2 by sqrt(time)
f5b <- FGR(Hist(time,cause)~cov2(X1,tf=qFun)+cov2(X2,tf=sqFun),data=d,cause=1)

## additional arguments for crr
f6<- FGR(Hist(time,cause)~X1+X2,data=d, cause=1,gtol=1e-5)
f6
f6a<- FGR(Hist(time,cause)~X1+X2,data=d, cause=1,gtol=0.1)
f6a
```

`Melanoma` *Malignant melanoma data*

Description

In the period 1962-77, 205 patients with malignant melanoma (cancer of the skin) had a radical operation performed at Odense University Hospital, Denmark. All patients were followed until the end of 1977 by which time 134 were still alive while 71 had died (of out whom 57 had died from cancer and 14 from other causes).

Format

A data frame with 205 observations on the following 12 variables.

time time in days from operation

status a numeric with values `0=censored` `1=death.malignant.melanoma` `2=death.other.causes`

event a factor with levels `censored` `death.malignant.melanoma` `death.other.causes`

invasion a factor with levels `level.0`, `level.1`, `level.2`

ici inflammatory cell infiltration (ICI): 0, 1, 2 or 3

epicel a factor with levels `not present` `present`

ulcer a factor with levels `not present` `present`

thick tumour thickness (in 1/100 mm)

sex a factor with levels `Female` `Male`

age age at operation (years)

logthick tumour thickness on log-scale

Details

The object of the study was to assess the effect of risk factors on survival. Among such risk factors were the sex and age of the patients and the histological variables tumor thickness and ulceration (absent vs. present).

Source

http://192.38.117.59/~linearpredictors/?page=datasets&dataset=Melanoma

References

Regression with linear predictors (2010)

Andersen, P.K. and Skovgaard, L.T.

Springer Verlag

Examples

```
data(Melanoma)
```

---

plot.riskRegression *Plotting predicted risk*

Description

Show predicted risk obtained by a risk prediction model as a function of time.

Usage

```
## S3 method for class 'riskRegression'
plot(x,
  cause,
  newdata,
  xlab,
  ylab,
  xlim,
  ylim,
  lwd,
  col,
  lty,
  axes=TRUE,
  percent=TRUE,
  legend=TRUE,
  add=FALSE,
  ...)
## S3 method for class 'CauseSpecificCox'
plot(x,
  cause,
  newdata,
  xlab,
  ylab,
  xlim,
  ylim,
```

```
  lwd,
  col,
  lty,
  axes=TRUE,
  percent=TRUE,
  legend=TRUE,
  add=FALSE,
  ...)
## S3 method for class 'predictedRisk'
plot(x,
  cause,
  newdata,
  xlab,
  ylab,
  xlim,
  ylim,
  lwd,
  col,
  lty,
  axes=TRUE,
  percent=TRUE,
  legend=TRUE,
  add=FALSE,
  ...)
```

Arguments

x	Fitted object obtained with one of ARR, LRR, riskRegression.
cause	For CauseSpecificCox models the cause of interest.
newdata	A data frame containing predictor variable combinations for which to compute predicted risk.
xlab	See plot
ylab	See plot
xlim	See plot
ylim	See plot
lwd	A vector of line thicknesses for the regression coefficients.
col	A vector of colors for the regression coefficients.
lty	A vector of line types for the regression coefficients.
axes	Logical. If FALSE then do not draw axes.
percent	If true the y-axis is labeled in percent.
legend	If true draw a legend.
add	Logical. If TRUE then add lines to an existing plot.
...	Used for transclusion of smart arguments for plot, lines, axis and background. See function SmartControl from prodlim.

Author(s)

Thomas Alexander Gerds <tag@biostat.ku.dk>

Examples

```
library(pec)
library(survival)
library(prodlim)
data(Melanoma)

fit.arr <- ARR(Hist(time,status)~invasion+age+strata(sex),data=Melanoma,cause=1)
plot(fit.arr)

fit.csc <- CSC(Hist(time,status)~invasion+age+sex,data=Melanoma,cause=1)
plot(fit.csc)
```

---

plotEffects *Plotting time-varying effects from a risk regression model.*

Description

Plot time-varying effects from a risk regression model.

Usage

```
plotEffects(x, formula, level, refLine = TRUE, confint = 0.95, xlim, ylim,
  xlab = "Time", ylab = "Cumulative coefficient", col, lty, lwd,
  add = FALSE, legend, axes = TRUE, ...)
```

Arguments

x	Fitted object obtained with one of ARR, LRR, riskRegression.
formula	A formula to specify the variable(s) whose regression coefficients should be plotted.
level	For categorical variables the level (group) whose contrast to the reference level (group) should be plotted.
refLine	Logical. If TRUE then add a horizontal line at zero.
confint	Logical. If TRUE then add confidence limits. Can be controlled using smart arguments. See examples
xlim	See plot
ylim	See plot
xlab	See plot
ylab	See plot
col	A vector of colors for the regression coefficients.
lty	A vector of line types for the regression coefficients.

lwd	A vector of line thicknesses for the regression coefficients.
add	Logical. If TRUE then add lines to an existing plot.
legend	Logical. If TRUE then add a legend. Can be controlled using smart arguments. See examples.
axes	Logical. If FALSE then do not draw axes.
...	Used for transclusion of smart arguments for plot, axis. See function SmartControl from prodlim.

Author(s)

Thomas H. Scheike <ts@biostat.ku.dk>

Thomas A. Gerds <tag@biostat.ku.dk>

Examples

```
library(pec)
library(survival)
library(prodlim)
data(Melanoma)

fit.tarr <- ARR(Hist(time,status)~strata(sex),
                data=Melanoma,
                cause=1)
plotEffects(fit.tarr)

fit.tarr <- ARR(Hist(time,status)~strata(sex)+strata(invasion),
                data=Melanoma,
                cause=1,
                times=seq(800,3000,20))
plotEffects(fit.tarr,formula=~sex)
plotEffects(fit.tarr,formula=~invasion)
plotEffects(fit.tarr,
            formula=~invasion,
            level="invasionlevel.1")

## legend arguments are transcluded:
plotEffects(fit.tarr,
            formula=~invasion,
            legend.bty="b",
            legend.cex=1)

## and other smart arguments too:
plotEffects(fit.tarr,
    formula=~invasion,
    legend.bty="b",
axis2.las=2,
    legend.cex=1)
```

predict.riskRegression *Predict individual risk.*

Description

Extract predictions from a risk prediction model.

Usage

```
## S3 method for class 'riskRegression'
predict(object, newdata, ...)
```

Arguments

object	Fitted object obtained with one of ARR, LRR, riskRegression.
newdata	A data frame containing predictor variable combinations for which to compute predicted risk.
...	not used

Author(s)

Thomas H. Scheike <ts@biostat.ku.dk>

Thomas A. Gerds <tag@biostat.ku.dk>

References

Gerds, TA and Scheike, T and Andersen, PK (2011) Absolute risk regression for competing risks: interpretation, link functions and prediction Research report 11/8. Department of Biostatistics, University of Copenhagen

Examples

```
data(Melanoma)
library(prodlim)
library(survival)

fit.tarr <- ARR(Hist(time,status)~age+invasion+strata(sex),data=Melanoma,cause=1)
predict(fit.tarr,newdata=data.frame(age=48,
                     invasion=factor("level.1",
                         levels=levels(Melanoma$invasion)),
                     sex=factor("Female",levels=levels(Melanoma$sex))))
predict(fit.tarr,newdata=data.frame(age=48,
                     invasion=factor("level.1",
                         levels=levels(Melanoma$invasion)),
                     sex=factor("Male",levels=levels(Melanoma$sex))))
predict(fit.tarr,newdata=data.frame(age=c(48,58,68),
                     invasion=factor("level.1",
```

```
                        levels=levels(Melanoma$invasion)),
                      sex=factor("Male",levels=levels(Melanoma$sex))))
predict(fit.tarr,newdata=Melanoma[1:4,])
```

`riskRegression` *Risk Regression Fits a regression model for the risk of an event – allowing for competing risks.*

Description

This is the twin-sister of the function `comp.risk` from the timereg package.

Usage

```
riskRegression(formula, data, times, link = "relative", cause,
  confint = TRUE, cens.model, cens.formula, numSimu = 0, maxiter = 50,
  silent = 1, convLevel = 6, conservative = TRUE, ...)
```

Arguments

`formula`	Formula where the left hand side specifies the event history event.history and the right hand side the linear predictor. See examples.
`data`	The data for fitting the model in which includes all the variables included in formula.
`times`	Vector of times. For each time point in `times` estimate the baseline risk and the timevarying coefficients.
`link`	`"relative"` for the absolute risk regression model. `"logistic"` for the logistic risk regression model. `"prop"` for the Fine-Gray regression model.
`cause`	The cause of interest.
`confint`	If `TRUE` return the iid decomposition, that can be used to construct confidence bands for predictions.
`cens.model`	Specified the model for the (conditional) censoring distribution used for deriving weights (ICPW). Defaults to "KM" (the Kaplan-Meier method ignoring covariates) alternatively it may be "Cox" (Cox regression).
`cens.formula`	Right hand side of the formula used for fitting the censoring model. If not specified the right hand side of `formula` is used.
`numSimu`	Number of simulations in resampling.
`maxiter`	Maximal number of iterations.
`silent`	Set this to `0` to see some system messages during fitting.
`convLevel`	Integer between 1 and 10, specifying the convergence criterion for the fitting algorithm as `10^{-convLevel}`.
`conservative`	If `TRUE` use variance formula that ignores the contribution by the estimate of the inverse of the probability of censoring weights
`...`	Not used.

Author(s)

Thomas H. Scheike <ts@biostat.ku.dk> Thomas A. Gerds <tag@biostat.ku.dk>

References

Gerds, TA and Scheike, T and Andersen, PK (2011) Absolute risk regression for competing risks: interpretation, link functions and prediction Research report 11/8. Department of Biostatistics, University of Copenhagen

Scheike, Zhang and Gerds (2008), Predicting cumulativeincidence probability by direct binomial regression,Biometrika, 95, 205-220.

Scheike and Zhang (2007), Flexible competing risks regression modelling and goodness of fit, LIDA, 14, 464-483.

Martinussen and Scheike (2006), Dynamic regression models for survival data, Springer.

Examples

```
data(Melanoma,package="riskRegression")
## tumor thickness on the log-scale
Melanoma$logthick <- log(Melanoma$thick)

# Single binary factor

## absolute risk regression
library(survival)
library(prodlim)
fit.arr <- ARR(Hist(time,status)~sex,data=Melanoma,cause=1)
print(fit.arr)
# show predicted cumulative incidences
plot(fit.arr,col=3:4,newdata=data.frame(sex=c("Female","Male")))

## compare with non-parametric Aalen-Johansen estimate
library(prodlim)
fit.aj <- prodlim(Hist(time,status)~sex,data=Melanoma)
plot(fit.aj,confint=FALSE)
plot(fit.arr,add=TRUE,col=3:4,newdata=data.frame(sex=c("Female","Male")))

## with time-dependent effect
fit.tarr <- ARR(Hist(time,status)~strata(sex),data=Melanoma,cause=1)
plot(fit.tarr,newdata=data.frame(sex=c("Female","Male")))

## logistic risk regression
fit.lrr <- LRR(Hist(time,status)~sex,data=Melanoma,cause=1)
summary(fit.lrr)


# Single continuous factor

## tumor thickness on the log-scale
Melanoma$logthick <- log(Melanoma$thick)
```

```
## absolute risk regression
fit2.arr <- ARR(Hist(time,status)~logthick,data=Melanoma,cause=1)
print(fit2.arr)
# show predicted cumulative incidences
plot(fit2.arr,col=1:5,newdata=data.frame(logthick=quantile(Melanoma$logthick)))

## comparison with nearest neighbor non-parametric Aalen-Johansen estimate
library(prodlim)
fit2.aj <- prodlim(Hist(time,status)~logthick,data=Melanoma)
plot(fit2.aj,confint=FALSE,newdata=data.frame(logthick=quantile(Melanoma$logthick)))
plot(fit2.arr,add=TRUE,col=1:5,lty=3,newdata=data.frame(logthick=quantile(Melanoma$logthick)))

## logistic risk regression
fit2.lrr <- LRR(Hist(time,status)~logthick,data=Melanoma,cause=1)
summary(fit2.lrr)

## change model for censoring weights
library(rms)
fit2a.lrr <- LRR(Hist(time,status)~logthick,
                 data=Melanoma,
                 cause=1,
                 cens.model="cox",
                 cens.formula=~sex+epicel+ulcer+age+logthick)
summary(fit2a.lrr)

##  compare prediction errors
## Not run:
    library(pec)
    plot(pec(list(ARR=fit2.arr,AJ=fit2.aj,LRR=fit2.lrr),data=Melanoma,maxtime=3000))

## End(Not run)


# multiple regression
library(pec)
library(riskRegression)
library(prodlim)
# absolute risk model
multi.arr <- ARR(Hist(time,status)~logthick+sex+age+ulcer,data=Melanoma,cause=1)

# stratified model allowing different baseline risk for the two gender
multi.arr <- ARR(Hist(time,status)~thick+strata(sex)+age+ulcer,data=Melanoma,cause=1)

# stratify by a continuous variable: strata(age)
multi.arr <- ARR(Hist(time,status)~tp(thick,power=0)+strata(age)+sex+ulcer,
                 data=Melanoma,
                 cause=1)

fit.arr2a <- ARR(Hist(time,status)~tp(thick,power=1),data=Melanoma,cause=1)
summary(fit.arr2a)
fit.arr2b <- ARR(Hist(time,status)~timevar(thick),data=Melanoma,cause=1)
summary(fit.arr2b)
```

```
## logistic risk model
fit.lrr <- LRR(Hist(time,status)~thick,data=Melanoma,cause=1)
summary(fit.lrr)




## nearest neighbor non-parametric Aalen-Johansen estimate
library(prodlim)
fit.aj <- prodlim(Hist(time,status)~thick,data=Melanoma)
plot(fit.aj,confint=FALSE)

## Not run:
    # prediction error
    library(pec)
    x <- pec(list(fit.arr2a,fit.arr2b,fit.lrr),
             data=Melanoma,
             formula=Hist(time,status)~1,
             cause=1,
             B=10,
             splitMethod="none")

## End(Not run)
```

`subjectWeights` *Estimation of censoring probabilities at subject specific times*

Description

This function is used internally to contruct pseudo values by inverse of the probability of censoring weights.

Usage

```
subjectWeights(formula, data, method = c("cox", "marginal", "km", "nonpar",
  "forest", "none"), args, lag = 1)
```

Arguments

`formula`	A survival formula like, Surv(time,status)~1 or Hist(time,status)~1 where status=0 means censored. The status variable is internally reversed for estimation of censoring rather than survival probabilities. Some of the available models, see argument `model`, will use predictors on the right hand side of the formula.
`data`	The data used for fitting the censoring model
`method`	Censoring model used for estimation of the (conditional) censoring distribution.
`args`	Arguments passed to the fitter of the method.
`lag`	If equal to `1` then obtain `G(T_i-|X_i)`, if equal to `0` estimate the conditional censoring distribution at the subjectTimes, i.e. (`G(T_i|X_i)`).

Details

Inverse of the probability of censoring weights usually refer to the probabilities of not being censored at certain time points. These probabilities are also the values of the conditional survival function of the censoring time given covariates. The function subjectWeights estimates the conditional survival function of the censoring times and derives the weights.

IMPORTANT: the data set should be ordered, `order(time,-status)` in order to get the `weights` in the right order for some choices of `method`.

Value

`times`	The times at which weights are estimated
`weights`	Estimated weights at individual time values `subjectTimes`
`lag`	The time lag.
`fit`	The fitted censoring model
`method`	The method for modelling the censoring distribution
`call`	The call

Author(s)

Thomas A. Gerds <tag@biostat.ku.dk>

Examples

```
library(pec)
library(prodlim)
library(survival)
dat=SimSurv(300)

dat <- dat[order(dat$time,-dat$status),]

# using the marginal Kaplan-Meier for the censoring times

WKM=subjectWeights(Hist(time,status)~X2,data=dat,method="marginal")
plot(WKM$fit)
WKM$fit
WKM$weights

# using the Cox model for the censoring times given X2

WCox=subjectWeights(Surv(time,status)~X2,data=dat,method="cox")
WCox
plot(WCox$weights,WKM$weights)

# using the stratified Kaplan-Meier for the censoring times given X2

WKM2 <- subjectWeights(Surv(time,status)~X2,data=dat,method="nonpar")
plot(WKM2$fit,add=FALSE)
```

Index