Package 'time2event'

July 30, 2016

Type Package

Title Survival and Competing Risk Analyses with Time-to-Event Data as Covariates

Version 0.1.0

Author Seongho Kim

Maintainer Seongho Kim <biostatistician.kim@gmail.com>

Depends survival, timereg

Description
Cox proportional hazard and competing risk regression analyses can be performed with time-to-event data as covariates.

License GPL-2

LazyData TRUE

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-30 09:30:37

R topics documented:

time2event-package *Survival and competing risk analyses with time-to-event data as covariates.*

Description

Cox proportional hazard and competing risk regression analyses can be performed with time-to-event data as covariates.

Details

Package:	time2event
Type:	Package
Version:	1.0
Date:	2016-7-27
License:	GPL-2

Author(s)

Seongho Kim <biostatistician.kim@gmail.com>

References

S. Kim (2016). time2event: an R package for the analysis of event time data with time-to-event data as covariates. Wayne State University/Karmanos Cancer Institute. Manuscript.

Examples

```
data(pegvhd)

# convert to data with time-to-event data as covariates
# os with cgvhd
tos1data = time2data(c("os.t","os.s"),c("gvhd.t","gvhd.s","pe.t","pe.s"),pegvhd)$data

# no time-varying analysis with 'coxph' and 'comp.risk'
os1r = coxph(Surv(os.t,os.s)~gvhd.s+pe.s+age+sex,data=pegvhd)

# time-varying analysis with 'coxph' and 'comp.risk'
nt.os1r = coxph(Surv(start,end,os.s)~gvhd.s+pe.s+age+sex,data=tos1data)

# time-varying analysis with 'tcoxph' and 'tcomp.risk'
t.os1r = tcoxph(Surv(os.t,os.s)~time(gvhd.t,gvhd.s)+time(pe.t,pe.s)+age+sex
 ,data=pegvhd)
```

```
data(bmtelder)

# convert to data with time-to-event data as covariates
# os with cgvhd
tos2data = time2data(c("os.t","os.s"),c("cgvhd.t","cgvhd.s"),bmtelder)$data

# nrm with cgvhd
tnrm2data = time2data(c("nrm.t","nrm.s"),c("cgvhd.t","cgvhd.s"),bmtelder)$data

# no time-varying analysis with 'coxph' and 'comp.risk'
os2r = coxph(Surv(os.t,os.s)~cgvhd.s+cond+donor,data=bmtelder)
set.seed(3927)
cr2r = comp.risk(Event(nrm.t,nrm.s)~cgvhd.s+cond+donor,data=bmtelder,
 cause=1,resample.iid=1,n.sim=1000,model="additive")
cr2r.pred = predict(cr2r,X=1)
plot(cr2r.pred)

# time-varying analysis with 'coxph' and 'comp.risk'
nt.os2r = coxph(Surv(start,end,os.s)~cgvhd.s+cond+donor,data=tos2data)
set.seed(3927)
nt.cr2r = comp.risk(Event(start,end,nrm.s)~cgvhd.s+cond+donor,data=tnrm2data,
 cause=1,resample.iid=1,n.sim=1000,model="additive")
nt.cr2r.pred = predict(nt.cr2r,X=1)
plot(nt.cr2r.pred)

# time-varying analysis with 'tcoxph' and 'tcomp.risk'
t.os2r = tcoxph(Surv(os.t,os.s)~time(cgvhd.t,cgvhd.s)+cond+donor,data=bmtelder)
set.seed(3927)
t.cr2r = tcomp.risk(Event(nrm.t,nrm.s)~time(cgvhd.t,cgvhd.s)+cond+donor,data=bmtelder,
 cause=1,resample.iid=1,n.sim=1000,model="additive")
t.cr2r.pred = predict(t.cr2r,X=1)
plot(t.cr2r.pred)
```

`bmtelder` *Bone marror transplant for old patients*

Description

This hypothetical data set gives the performance of allogenic bone marror transplantation for old patients. This data set is provided to illustrate the analysis with time-to-event data as covariates, in particular, competing risk regression.

Usage

```
bmtelder
```

Format

This data frame consists of 50 individuals and the following columns:

- os.tTime from the date of transplantation to the date of death. Days
- os.sStatus. 1 = death and 0 = censored
- nrm.tTime from the date of transplantation to the date of death without relapse, with relapse as competing risk. Days
- nrm.sStatus. 2 = relapse, 1 = death, and 0 = censored
- cgvhd.tTime from the date of transplantation to the date of chronic graft-verse-host disease (GVHD). Days
- cgvhd.sStatus. 1 = chronic GVHD and 0 = censored
- condConditioning regimen. Reduced intensity and full intensity
- donordonor to host matching. Allo Related (allogeneic related) and Allo Unrelated (allogeneic unrelated)

---

pegvhd *Plueral effusion and graft-verse-host disease after transplantation*

Description

This hypothetical data set is the study for the plueral effusion and graft-verse-host disease (GVHD) after allogenic transplantation. This data set is provided to illustrate the analysis with time-to-event data as covariates.

Usage

```
pegvhd
```

Format

This data frame consists of 50 individuals (25 with plueral effusion and 25 without plueral effusion) and the following columns:

- os.tTime from the date of transplantation to the date of death. Days
- os.sStatus. 1 = death and 0 = censored
- gvhd.tTime from the date of transplantation to the date of GVHD (either acute GVHD or chronic GVHD, whichever is first). Days
- gvhd.sStatus. 1 = GVHD and 0 = censored
- pe.tTime from the date of transplantation to the date of plueral effusion. Days
- pe.sStatus. 1 = plueral effusion and 0 = censored
- ageAge
- sexGender

References

Modi D, Jang H, Kim S, Deol A, Ayash L, Bhutani D, Lum LG, Ratanatharathorn V, Manasa R, Mellert K, Uberti JP. (2016) Incidence, etiology, and outcome of pleural effusions in allogeneic hematopoietic stem cell transplantation. *American Journal of Hematology*, in press.

`tcomp.risk` *Competings Risks Regression with time-to-event data as covariates.*

Description

Fits a semiparametric model for the cause-specific quantitie with time-to-event data as covariates.

Usage

```
tcomp.risk(formula, na.time=c("remove","censor"), verbose=FALSE,
      data = sys.parent(), cause, times = NULL,
     Nit = 50, clusters = NULL, est = NULL, fix.gamma = 0, gamma = 0,
     n.sim = 0, weighted = 0, model = "fg", detail = 0, interval = 0.01,
     resample.iid = 1, cens.model = "KM", cens.formula = NULL,
     time.pow = NULL, time.pow.test = NULL, silent = 1, conv = 1e-06,
     weights = NULL, max.clust = 1000, n.times = 50, first.time.p = 0.05,
     estimator = 1, trunc.p = NULL, cens.weights = NULL, admin.cens = NULL,
     conservative = 1, monotone = 0, step = NULL)
```

Arguments

`formula`	a formula object, with the response on the left of a '~' operator, and the terms on the right. The response must be a survival object as returned by the 'Event' function. The status indicator is not important here. Time-invariant regressors are specified by the wrapper `const()`, and cluster variables (for computing robust variances) by the wrapper `cluster()`. In case that time-to-event data are covariates, use the wrapper `time()` to indicate the time-to-event data as covariates.
`na.time`	a missing-data filter function for time-to-event covariates. The option 'remove' will remove all the data with 'NA', while the option 'censor' will treat the missing data as censored and then replace with the logest time. Default is 'remove'.
`verbose`	logical. Should R report extra information on progress? Default is 'FALSE'.
`data`	a data.frame with the variables.
`cause`	For competing risk models specificies which cause we consider.
`times`	specifies the times at which the estimator is considered. Defaults to all the times where an event of interest occurs, with the first 10 percent or max 20 jump points removed for numerical stability in simulations.
`Nit`	number of iterations for Newton-Raphson algorithm.
`clusters`	specifies cluster structure, for backwards compability.

est	possible starting value for nonparametric component of model.
fix.gamma	to keep gamma fixed, possibly at 0.
gamma	starting value for constant effects.
n.sim	number of simulations in resampling.
weighted	Not implemented. To compute a variance weighted version of the test-processes used for testing time-varying effects.
model	"additive", "prop"ortional, "rcif", or "logistic".
detail	if 0 no details are printed during iterations, if 1 details are given.
interval	specifies that we only consider timepoints where the Kaplan-Meier of the censoring distribution is larger than this value.
resample.iid	to return the iid decomposition, that can be used to construct confidence bands for predictions
cens.model	specified which model to use for the ICPW, KM is Kaplan-Meier alternatively it may be "cox"
cens.formula	specifies the regression terms used for the regression model for chosen regression model. When cens.model is specified, the default is to use the same design as specified for the competing risks model.
time.pow	specifies that the power at which the time-arguments is transformed, for each of the arguments of the const() terms, default is 1 for the additive model and 0 for the proportional model.
time.pow.test	specifies that the power the time-arguments is transformed for each of the arguments of the non-const() terms. This is relevant for testing if a coefficient function is consistent with the specified form $A_l(t) = beta_l t^t ime.pow.test(l)$. Default is 1 for the additive model and 0 for the proportional model.
silent	if 0 information on convergence problems due to non-invertible derviates of scores are printed.
conv	gives convergence criterie in terms of sum of absolute change of parameters of model
weights	weights for estimating equations.
max.clust	sets the total number of i.i.d. terms in i.i.d. decompostition. This can limit the amount of memory used by coarsening the clusters. When NULL then all clusters are used. Default is 1000 to save memory and time.
first.time.p	first point for estimation is pth percentile of cause jump times.
n.times	only uses 50 points for estimation, if NULL then uses all points, subject to p.start condition.
estimator	default estimator is 1.
trunc.p	truncation weight for delayed entry, P(T > entry.time | Z_i), typically Cox model.
cens.weights	censoring weights can be given here rather than calculated using the KM, cox or aalen models.
admin.cens	censoring times for the administrative censoring
conservative	set to 0 to compute correct variances based on censoring weights, default is conservative estimates that are much quicker.
monotone	monotone=0, uses estimating equations montone 1 uses
step	step size for Fisher-Scoring algorithm.

Details

The funciton `tcomp.risk` is an extention of the function `comp.risk` for time-to-event covariates. If the model has no time-to-event covariates, `tcomp.risk` will print the warning sign 'No time-varying covariate!!!' and then do exactly the same procedure as `comp.risk` does. If the model has time-to-event covariates, the time-to-event covaraites should be wrapped with `time()` by placing the right-hand side of a ~ operator. In particular, the wrapper `time(a1,b1,a2,b2,a3,b3,...)` will be used with time-to-event covariates, where ai and bi, $i = 1, 2, ...$ are time-to-event and status, respectively. See `comp.risk` for other details.

Value

returns the same object as that of `comp.risk()`. See `comp.risk()` for details

Author(s)

Seongho Kim

References

S. Kim (2016). time2event: an R package for the analysis of event time data with time-to-event data as covariates. Wayne State University/Karmanos Cancer Institute. Manuscript.

Examples

```
data(bmtelder)

# convert to data with time-to-event data as covariates
# nrm with cgvhd
tnrm2data = time2data(c("nrm.t","nrm.s"),c("cgvhd.t","cgvhd.s"),bmtelder)$data

# no time-varying analysis with 'comp.risk'
set.seed(3927)
cr2r = comp.risk(Event(nrm.t,nrm.s)~cgvhd.s+cond+donor,data=bmtelder,
 cause=1,resample.iid=1,n.sim=1000,model="additive")
cr2r.pred = predict(cr2r,X=1)
plot(cr2r.pred)

# time-varying analysis with 'comp.risk'
set.seed(3927)
nt.cr2r = comp.risk(Event(start,end,nrm.s)~cgvhd.s+cond+donor,data=tnrm2data,
 cause=1,resample.iid=1,n.sim=1000,model="additive")
nt.cr2r.pred = predict(nt.cr2r,X=1)
plot(nt.cr2r.pred)

# time-varying analysis with 'tcomp.risk'
set.seed(3927)
t.cr2r = tcomp.risk(Event(nrm.t,nrm.s)~time(cgvhd.t,cgvhd.s)+cond+donor,data=bmtelder,
 cause=1,resample.iid=1,n.sim=1000,model="additive")
t.cr2r.pred = predict(t.cr2r,X=1)
plot(t.cr2r.pred)
```

tcoxph *Fit Proportional Hazards Regression Model with time-to-event data as covariates.*

Description

Fits a Cox proportional hazards regression model with time-to-event data as covariates.

Usage

```
tcoxph(formula, na.time=c("remove","censor"),verbose=FALSE,
                    data, weights, subset, na.action, init, control,
                    ties = c("efron", "breslow", "exact"), singular.ok = TRUE,
                  robust, model = FALSE, x = FALSE, y = TRUE, tt, method = ties,
                    ...)
```

Arguments

formula	a formula object, with the response on the left of a ~ operator, and the terms on the right. The response must be a survival object as returned by the Surv function. In case that time-to-event data are covariates, use the wrapper `time()` to indicate the time-to-event data as covariates.
na.time	a missing-data filter function for time-to-event covariates. The option 'remove' will remove all the data with 'NA', while the option 'censor' will treat the missing data as censored and then replace with the logest time. Default is 'remove'.
verbose	logical. Should R report extra information on progress? Default is 'FALSE'.
data	a data.frame in which to interpret the variables named in the formula, or in the subset and the weights argument.
weights	vector of case weights. For a thorough discussion of these see the book by Therneau and Grambsch.
subset	expression indicating which subset of the rows of data should be used in the fit. All observations are included by default.
na.action	a missing-data filter function. This is applied to the model.frame after any subset argument has been used. Default is `options()`\$na.action.
init	vector of initial values of the iteration. Default initial value is zero for all variables.
control	Object of class coxph.control specifying iteration limit and other control options. Default is `coxph.control(...)`.
ties	a character string specifying the method for tie handling. If there are no tied death times all the methods are equivalent. Nearly all Cox regression programs use the Breslow method by default, but not this one. The Efron approximation is used as the default here, it is more accurate when dealing with tied death times, and is as efficient computationally. The "exact partial likelihood" is equivalent to a conditional logistic model, and is appropriate when the times are a small set of discrete values. See further below.

singular.ok	logical value indicating how to handle collinearity in the model matrix. If TRUE, the program will automatically skip over columns of the X matrix that are linear combinations of earlier columns. In this case the coefficients for such columns will be NA, and the variance matrix will contain zeros. For ancillary calculations, such as the linear predictor, the missing coefficients are treated as zeros.
robust	this argument has been deprecated, use a cluster term in the model instead. (The two options accomplish the same goal – creation of a robust variance – but the second is more flexible).
model	logical value: if TRUE, the model frame is returned in component model.
x	logical value: if TRUE, the x matrix is returned in component x.
y	logical value: if TRUE, the response vector is returned in component y.
tt	optional list of time-transform functions.
method	alternate name for the ties argument.
...	Other arguments will be passed to coxph.control

Details

The funciton tcoxph is an extention of the function coxph for time-to-event covariates. If the model has no time-to-event covariates, tcoxph will print the warning sign 'No time-varying covariate!!!' and then do exactly the same procedure as coxph does. If the model has time-to-event covariates, the time-to-event covaraites should be wrapped with time() by placing the right-hand side of a ~ operator. In particular, the wrapper time(a1,b1,a2,b2,a3,b3,...) will be used with time-to-event covariates, where ai and bi, $i = 1, 2, ...$ are time-to-event and status, respectively. See coxph for other details.

Value

the same object as that of class coxph representing the fit. See coxph.object for details.

Author(s)

Seongho Kim

References

S. Kim (2016). time2event: an R package for the analysis of event time data with time-to-event data as covariates. Wayne State University/Karmanos Cancer Institute. Manuscript.

Examples

```
data(pegvhd)

# convert to data with time-to-event data as covariates
# os with cgvhd
tos1data = time2data(c("os.t","os.s"),c("gvhd.t","gvhd.s","pe.t","pe.s"),pegvhd)$data

# no time-varying analysis with 'coxph' and 'comp.risk'
os1r = coxph(Surv(os.t,os.s)~gvhd.s+pe.s+age+sex,data=pegvhd)
```

```
# time-varying analysis with 'coxph' and 'comp.risk'
nt.os1r = coxph(Surv(start,end,os.s)~gvhd.s+pe.s+age+sex,data=tos1data)

# time-varying analysis with 'tcoxph' and 'tcomp.risk'
t.os1r = tcoxph(Surv(os.t,os.s)~time(gvhd.t,gvhd.s)+time(pe.t,pe.s)+age+sex
 ,data=pegvhd)

data(bmtelder)

# convert to data with time-to-event data as covariates
# os with cgvhd
tos2data = time2data(c("os.t","os.s"),c("cgvhd.t","cgvhd.s"),bmtelder)$data

# no time-varying analysis with 'coxph'
os2r = coxph(Surv(os.t,os.s)~cgvhd.s+cond+donor,data=bmtelder)

# time-varying analysis with 'coxph'
nt.os2r = coxph(Surv(start,end,os.s)~cgvhd.s+cond+donor,data=tos2data)

# time-varying analysis with 'tcoxph'
t.os2r = tcoxph(Surv(os.t,os.s)~time(cgvhd.t,cgvhd.s)+cond+donor,data=bmtelder)
```

`time2data` *Classification-based quantitative SILAC analysis*

Description

finding differentially expressed proteins using classification methods

Usage

```
time2data(tvar,tcov,data,na.time=c("remove","censor"),verbose=FALSE,weights=NULL)
```

Arguments

`tvar`	a vector of time-to-event response variables' names as a pair of 'time to event' and 'status'.
`tcov`	a vector of time-to-event covariates' names as a pair of 'time to event' and 'status'.
`data`	a data.frame in which to interpret the variables named in the formula, or in the subset and the weights argument. The variables in `tvar` and `tcov` must be part of data.
`na.time`	a missing-data filter function for time-to-event covariates. The option 'remove' will remove all the data with 'NA', while the option 'censor' will treat the missing data as censored and then replace with the logest time. Default is 'remove'.
`verbose`	logical. Should R report extra information on progress? Default is 'FALSE'.

`weights` vector of case weights. For a thorough discussion of these see the book by Therneau and Grambsch. Default is 'NULL'.

Details

It will restructure the data according to time-to-event covariates. After converting, the time-to-event response variable will be formulated as interval censored data and its structure will be 'start', 'end', and 'status'.

Value

`data` restructured data by time-to-event data as covariates. The time-to-event response variable will be structured as interval censored data as the form of 'start','end', and 'status'.

`wt` restructed weights

Author(s)

Seongho Kim

References

S. Kim (2016). time2event: an R package for the analysis of event time data with time-to-event data as covariates. Wayne State University/Karmanos Cancer Institute. Manuscript.

Examples

```
data(pegvhd)

# convert to data with time-to-event data as covariates
# os with cgvhd
tos1data = time2data(c("os.t","os.s"),c("gvhd.t","gvhd.s","pe.t","pe.s"),pegvhd)$data

data(bmtelder)

# convert to data with time-to-event data as covariates
# os with cgvhd
tos2data = time2data(c("os.t","os.s"),c("cgvhd.t","cgvhd.s"),bmtelder)$data

# nrm with cgvhd
tnrm2data = time2data(c("nrm.t","nrm.s"),c("cgvhd.t","cgvhd.s"),bmtelder)$data
```

Index