

Package ‘rcure’

January 21, 2017

Type Package

Title Robust Cure Models for Survival Analysis

Version 0.1.0

Description Implements robust cure models for survival analysis by incorporate a weakly informative prior in the logistic part of cure models. Estimates prognostic accuracy, i.e. AUC, k-index and c-index, with bootstrap confidence interval for cure models.

Depends R (>= 3.3.1)

License GPL-3

Encoding UTF-8

LazyData true

Imports smcure, arm, survival, MASS, plyr

RoxygenNote 5.0.1

NeedsCompilation no

Author Xiaoxia Han [aut, cre],
Yilong Zhang [aut],
Yongzhao Shao [aut]

Maintainer Xiaoxia Han <xh414@nyu.edu>

Repository CRAN

Date/Publication 2017-01-21 02:07:47

R topics documented:

coefrcure	2
eva_cure	2
eva_rcure	3
plotpredictrcure	3
predictrcure	4
printrcure	4
rcure	5

Index	8
--------------	----------

coefrcure	<i>Retrieves the estimated coefficients from rcure object</i>
-----------	---

Description

Retrieves the estimated coefficients from rcure object

Usage

```
coefrcure(x, ...)
```

Arguments

x	an object of rcure
...	further arguments to be passed to the coefrcure function

References

Cai, C., Zou, Y., Peng, Y., & Zhang, J. (2012). smcure: An R-Package for estimating semiparametric mixture cure models. *Computer methods and programs in biomedicine*, 108(3), 1255-1260

eva_cure	<i>Evaluate Mixture cure model fitted by rcure</i>
----------	--

Description

Evaluate Mixture cure model fitted by rcure

Usage

```
eva_cure(time, delta, X, beta, Z, b, surv, model, baseline = T)
```

Arguments

time	survival time
delta	event status (1 = event)
X	design matrix of survival part
beta	parameters of survival part
Z	design matrix of cure part
b	parameters of cure part
surv	survival probability
model	the type of survival model ("PH", "PO", "Normal")
baseline	whether the surv is baseline survival probability

eva_rcure	<i>Evaluate Mixture cure model fitted by rcure</i>
-----------	--

Description

Evaluate Mixture cure model fitted by rcure

Usage

```
eva_rcure(fit, model)
```

Arguments

fit	an extended smcure object
model	the type of survival model ("PH", "PO", "Normal")

plotpredictrcure	<i>Plots predicted rcure object</i>
------------------	-------------------------------------

Description

Plots predicted survival curve(s) from the estimated robust cure model

Usage

```
plotpredictrcure(object, type = "S", xlab = "Time",
  ylab = "Predicted Survival Probability", model = c("ph", "aft"), ...)
```

Arguments

object	an object of the predictrcure function
type	type of plot. "S" means steps plot.
xlab	label for the x axis
ylab	label for the y axis
model	specifies survival part in cure model, "ph" or "aft"
...	further arguments to be passed to the plotpredictrcure function

References

Cai, C., Zou, Y., Peng, Y., & Zhang, J. (2012). smcure: An R-Package for estimating semiparametric mixture cure models. *Computer methods and programs in biomedicine*, 108(3), 1255-1260

predictrcure *Prediction of robust cure model*

Description

Prediction of survival probability from rcure() in order to pass subjects to plotpredictrcure().

Usage

```
predictrcure(object, newX, newZ, model = c("ph", "aft"), ...)
```

Arguments

object	an object of rcure
newX	new value(s) of X
newZ	new value(s) of Z
model	specifies survival part in cure model, "ph" or "aft"
...	further arguments to be passed to the predictrcure function

References

Cai, C., Zou, Y., Peng, Y., & Zhang, J. (2012). smcure: An R-Package for estimating semiparametric mixture cure models. *Computer methods and programs in biomedicine*, 108(3), 1255-1260

printrcure *Prints rcure object*

Description

Prints rcure object

Usage

```
printrcure(x, Var = TRUE, ROC = TRUE, ...)
```

Arguments

x	an object of rcure
Var	if it is TRUE, the function returns standard error estimated by bootstrap method. If it is False, the function only returns estimators of coefficients. By default, Var = TRUE
ROC	if TRUE, the function plots ROC. By default, ROC = TRUE
...	further arguments to be passed to the printrcure function

References

Cai, C., Zou, Y., Peng, Y., & Zhang, J. (2012). smcure: An R-Package for estimating semiparametric mixture cure models. *Computer methods and programs in biomedicine*, 108(3), 1255-1260

 rcure

Robust cure model

Description

Fits robust cure model by incorporating a weakly informative prior distribution for uncure probability part in cure models

Usage

```
rcure(formula, cureform, offset.s = NULL, data, na.action = na.omit,
      model = c("aft", "ph"), link = "logit", Var = TRUE, emmax = 50,
      eps = 1e-07, nboot = 100, family = binomial(link = "logit"),
      method = "glm.fit", prior.mean = 0, prior.scale = NULL, prior.df = 1,
      prior.mean.for.intercept = 0, prior.scale.for.intercept = NULL,
      prior.df.for.intercept = 1, min.prior.scale = 1e-12, scaled = TRUE,
      n.iter = 100, Warning = TRUE, eva_model = NULL, cutpoint = c(0.1,
      0.25, 0.5, 0.75, 0.9))
```

Arguments

formula	a formula object for the survival part in cure model. left must be a survival object as returned by the Surv function
cureform	specifies the variables in the uncure probability part in cure model
offset.s	variable(s) with coefficient 1 in PH model or AFT model
data	a a.data.frame
na.action	a missing-data filter function. By default na.action = na.omit
model	specifies survival part in cure model, "ph" or "aft"
link	specifies the link in incidence part. The "logit", "probit" or complementary loglog ("cloglog") links are available. By default link = "logit".
Var	If it is TRUE, the program returns Std.Error by bootstrap method. If set to False, the program only returns estimators of coefficients. By default, Var = TRUE
emmax	specifies the maximum iteration number. If the convergence criterion is not met, the EM iteration will be stopped after emmax iterations and the estimates will be based on the last maximum likelihood iteration. The default emmax = 100.
eps	sets the convergence criterion. The default is eps = 1e-7. The iterations are considered to be converged when the maximum relative change in the parameters and likelihood estimates between iterations is less than the value specified.
nboot	specifies the number of bootstrap sampling. The default nboot = 100

family	a description of the error distribution and link function to be used in the model. Default is binomial(link="logit")
method	the method to be used in fitting the glmbayes model. The default method "glm.fit" uses iteratively reweighted least squares (IWLS). The only current alternative is "model.frame" which returns the model frame and does no fitting
prior.mean	prior mean for the coefficients: default is 0. Can be a vector of length equal to the number of predictors (not counting the intercept, if any). If it is a scalar, it is expanded to the length of this vector.
prior.scale	prior scale for the coefficients: default is NULL; if is NULL, for a logit model, prior.scale is 2.5; for a probit model, prior scale is 2.5*1.6. Can be a vector of length equal to the number of predictors (not counting the intercept, if any). If it is a scalar, it is expanded to the length of this vector
prior.df	prior degrees of freedom for the coefficients. For t distribution: default is 1 (Cauchy). Set to Inf to get normal prior distributions. Can be a vector of length equal to the number of predictors (not counting the intercept, if any). If it is a scalar, it is expanded to the length of this vector
prior.mean.for.intercept	prior mean for the intercept: default is 0.
prior.scale.for.intercept	prior scale for the intercept: default is NULL; for a logit model, prior scale for intercept is 10; for probit model, prior scale for intercept is rescaled as 10*1.6.
prior.df.for.intercept	prior degrees of freedom for the intercept: default is 1.
min.prior.scale	Minimum prior scale for the coefficients: default is 1e-12.
scaled	scaled=TRUE, the scales for the prior distributions of the coefficients are determined as follows: For a predictor with only one value, we just use prior.scale. For a predictor with two values, we use prior.scale/range(x). For a predictor with more than two values, we use prior.scale/(2*sd(x)). If the response is Gaussian, prior.scale is also multiplied by 2 * sd(y). Default is TRUE
n.iter	integer giving the maximal number of bayesglm IWLS iterations, default is 100.
Warning	default is TRUE, which will show the error messages of not convergence and separation
eva_model	for Cox PH model, the default is "PH". For AFT model, it can be "PO".
cutpoint	the cut points for ROC to calculate AUC
...	further arguments passed to or from other methods

Author(s)

Xiaoxia Han

References

- Cai, C., Zou, Y., Peng, Y., & Zhang, J. (2012). smcure: An R-Package for estimating semiparametric mixture cure models. *Computer methods and programs in biomedicine*, 108(3), 1255-1260.
- Gelman, A., Jakulin, A., Pittau, M. G., & Su, Y. S. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, 1360-1383.

Examples

```

library(survival)
library(smcurve)
library(arm)
data(e1684)

# fit PH robust cure model
pd <- rcure(Surv(FAILTIME,FAILCENS)~TRT+SEX+AGE,cureform=~TRT+SEX+AGE,
data=e1684,model="ph",Var =FALSE,
method = "glm.fit", prior.mean = 0, prior.scale = NULL, prior.df = 1,
prior.mean.for.intercept = 0, prior.scale.for.intercept = NULL,
prior.df.for.intercept = 1, min.prior.scale = 1e-12,
scaled = FALSE, n.iter = 100, Warning=F)
printrcure(pd,Var = FALSE, ROC=FALSE)
# plot predicted survival curves for male with median centered age by treatment groups
predm=predictrcure(pd,newX=cbind(c(1,0),c(0,0),c(0.579,0.579)),
newZ=cbind(c(1,0),c(0,0),c(0.579,0.579)),model="ph")
plotpredictrcure(predm,model="ph")

# just a test: this should be identical to classical cure model
pd2 <- rcure(Surv(FAILTIME,FAILCENS)~TRT+SEX+AGE,cureform=~TRT+SEX+AGE,
data=e1684,model="ph",Var = FALSE,
method = "glm.fit", prior.mean = 0, prior.scale = Inf, prior.df = Inf,
prior.mean.for.intercept = 0, prior.scale.for.intercept = Inf,
prior.df.for.intercept = Inf, Warning=F)
printrcure(pd2,Var = FALSE, ROC=FALSE)
pd3 <- smcurve(Surv(FAILTIME,FAILCENS)~TRT+SEX+AGE,cureform=~TRT+SEX+AGE,
data=e1684,model="ph",Var = FALSE)

data(bmt)
# fit AFT robust cure model
bmtfit <- rcure(formula = Surv(Time, Status) ~ TRT, cureform = ~TRT,
data = bmt, model = "aft", Var = FALSE,
method = "glm.fit", prior.mean = 0, prior.scale = NULL, prior.df = 1,
prior.mean.for.intercept = 0, prior.scale.for.intercept = NULL,
prior.df.for.intercept = 1, min.prior.scale = 1e-12,
scaled = TRUE, n.iter = 100, Warning=F, eva_mode="PO")
printrcure(bmtfit,Var = FALSE, ROC=FALSE)
## plot predicted Survival curves by treatment groups
predbmt=predictrcure(bmtfit,newX=c(0,1),newZ=c(0,1),model="aft")
plotpredictrcure(predbmt,model="aft")

```

Index

`coefrcure`, 2

`eva_cure`, 2

`eva_rcure`, 3

`plotpredictrcure`, 3

`predictrcure`, 4

`printrcure`, 4

`rcure`, 5