Package 'mixcure'

November 24, 2020

Encoding UTF-8

Title Mixture Cure Models

Version 2.0

Description Implementation of parametric and semiparametric mixture cure models based on existing R packages. See details of the models in Peng and Yu (2020) <ISBN: 9780367145576>.

Depends R (>= 3.5.0), survival

Imports boot, flexsurv, survey, gam, timereg

License GPL-3

RoxygenNote 7.1.0

NeedsCompilation no

Author Yingwei Peng [aut, cre]

Maintainer Yingwei Peng <yingwei.peng@queensu.ca>

Repository CRAN

Date/Publication 2020-11-24 12:20:02 UTC

R topics documented:

mixcure-package *Mixture cure models*

Description

Fit various mixture cure models

Details

This package fits various mixture cure model using existing R packages.

Author(s)

Yingwei Peng

References

Peng, Y. and Yu, B. Cure Models: Methods, Applications, and Implementation. CRC/Chapman & Hall, 2020

Peng, Y. and Taylor, J. M. G. Cure models. In Klein, J., van Houwelingen, H., Ibrahim, J. G., and Scheike, T. H., editors, Handbook of Survival Analysis, Handbooks of Modern Statistical Methods series, chapter 6, pages 113-134. Chapman & Hall, Boca Raton, FL, USA, 2014

Peng, Y. Fitting semiparametric cure models. Computational Statistics & Data Analysis, 41: 481-490, 2003

coef.mixcure *Retrieve coefficients from mixture cure models*

Description

Retrieve coefficients from mixture cure models

Usage

```
## S3 method for class 'mixcure'
coef(object, ...)
```

Arguments

object	a mixcure object
...	for compatibility purpose. Not used.

Value

a list of two components:

`latency`	a vector of coefficients in the latency model
`incidence`	a vector of coefficients in the incidence model

Author(s)

Yingwei Peng

See Also

mixcure

Examples

```
data(leukaemia)
z1 = mixcure(Surv(time, cens) ~ transplant, ~ transplant, data = leukaemia, savedata = TRUE)
coef(z1)
```

---

leukaemia *Data from leukaemia patients with bone marrow transplants*

Description

This data set is used in the examples of this package.

Usage

```
data(leukaemia)
```

Format

A data.frame object with columns:

`time`: Relapse Free Survival Time.

`cens`: Relapse Indicator with 1-Relapsed, 0-Disease Free.

`transplant`: Bone marrow transplant: 0-Allogeneic, 1-Autologous.

Source

Kersey JH, Weisdorf D, Nesbit ME, LeBien TW, Woods WG, McGlave PB, Kim TRUE, Vallera DA, Goldman AI, Bostrom B, Hurd D, Ramsay NKC. Comparison of autologous and allgeneic bone marrow transplantation for treatment of high-risk refractory acute lymphoblastic leukaemia. New England Journal of Medicine 1987; 317:461– 467.

Examples

```
data(leukaemia)
```

`mixcure` *Mixture cure models*

Description

Fit some parametric and semiparametric mixture cure models

Usage

```
mixcure(
  lformula,
  iformula,
  data,
  lmodel,
  imodel,
  postuncure = NULL,
  emmax = 100,
  eps = 1e-04,
  savedata = FALSE,
  debug = FALSE
)
```

Arguments

`lformula`	a formula specifying the latency model
`iformula`	a formula specifying the incidence model
`data`	a data frame in which to interpret the variables named in the formulas in lmodel and imodel.
`lmodel`	a list of at least one component: fun, a text string specifying R function to fit the latency model. Other arguments to the function specified in fun can be added to the list. If lmodel is not supplied, the default is lmodel = list(fun = "coxph") and it fits the semiparametric PH latency model to the data. Other latency models currently implemented include • fun = "survfit" (for nonparametric mixture cure models), • fun = "survreg" (for parametric mixture cure models), • fun = "flexsurvreg" (for parametric mixture cure models), • fun = "flexsurvspline" (for parametric mixture cure models), • fun = "cox.aalen" (for semiparametric multiplicative-additive hazards models), • fun = "prop.odds" (for semiparametric proportional odds models).

R functions for other regression models for survival data can be added as long as they meet certain criteria. See details on how a new regression model for survival data can be added.

`imodel` a list of at least one component: fun, a text string specifying the R function to fit the incidence model. Other arguments to the function specified in fun can be added to the list. If imodel is not supplied, the default is imodel = list(fun = "glm", family = binomial()) and it fits the logistic regression as the incidence model to the data. The other incidence model currently implemented is fun = "gam". R functions for other regression models as the incidence model can be added as long as they meet certain criteria. See details on how a new regression model for incidence model can be added.

A covariate may be used in both lformula and iformula. A model with the intercept term only in iformula assumes that there are cured patients and that the cure rates are the same for all patients.

`postuncure` a vector of initial probabilities of being uncured for all subjects

`emmax` the maximum number of EM iterations

`eps` tolerance for EM convergence. Iteration stops once the relative change in log likelihoods is less than eps.

`savedata` If TRUE, the data set will be stored in the final object. It is mainly used for bootstrap in `summary()` to get standard errors. Default is FALSE.

`debug` for debug purpose.

Details

This function fits mixture cure models, where the latency and incidence parts of the mixture cure models can be fit using existing R regression functions. We implemented the logistic regression and the generalized additive model for the incidence part and the semiparametric proportional hazards and additive hazards models and the parametric failure time models for the latency parts. To include a new regression model xxx() for the incidence part, you need to add the following functions:

- incidence.xxx()
- coef.incidence.xxx()
- loglik.incidence.xxx()
- curepred.incidence.xxx()

To include a new regression model xxx() for the latency part, you need to add the following functions:

- latency.xxx()
- coef.latency.xxx()
- loglik.latency.xxx()
- survpred.latency.xxx()

When mixcure program ends, it may produce warning messages such as "In eval(expr, envir, enclos) : non-integer #successes in a binomial glm!." The message is due to the fact that glm is picky when it comes to specifying binomial models. It warns if it detects that the number of trials or successes

is non-integer, but it still fits the model properly. If you want to suppress the warning (and you're sure it's not a problem), use family=quasibinomial instead.

This package requires the following R packages: survival, boot, survey and any packages that provide the latency and incidence models.

Value

a list consists of the following components:

`ifit`	the final fit from the model for cure probability.
`lfit`	the final fit from the model for failure time of uncured subjects
`survprob`	the final estimate of the survival probability for each subject at its own observed time if uncured
`postuncure`	the final estimate of the posterior uncure probability for each subject based its observed values. The prior uncure probability can be obtained from ifit$uncureprob
`em`	a matrix containing the iterations of the EM algorithm and convergence errors (for debug purpose)

Author(s)

Yingwei Peng

References

Peng, Y. and Yu, B. Cure Models: Methods, Applications, and Implementation. CRC/Chapman & Hall, 2020

Peng, Y. and Taylor, J. M. G. Cure models. In Klein, J., van Houwelingen, H., Ibrahim, J. G., and Scheike, T. H., editors, Handbook of Survival Analysis, Handbooks of Modern Statistical Methods series, chapter 6, pages 113-134. Chapman & Hall, Boca Raton, FL, USA, 2014

Peng, Y. Fitting semiparametric cure models. Computational Statistics & Data Analysis, 41: 481-490, 2003

See Also

summary.mixcure, predict.mixcure, plot.predict.mixcure

Examples

```
data(leukaemia)
z = mixcure(Surv(time, cens) ~ transplant, ~ transplant, data = leukaemia)
```

`plot.predict.mixcure` *Plot method for the prediction of mixture cure models*

Description

Plot the predicted survival function curves from mixture cure models

Usage

```
## S3 method for class 'predict.mixcure'
plot(
  x,
  type = "l",
  add = FALSE,
  which = 1:nrow(x$cure),
  curemark = FALSE,
  conditional = FALSE,
  xlab,
  ylab,
  ylim = c(0, 1),
  lty = seq(along = which),
  ...
)
```

Arguments

`x`	an object from predict.mixcure
`type`	line type. The default is type = "l"
`add`	if add = FALSE (default), a plot is shown in a new graphics window. Otherwise, an existing window is used.
`which`	a vector of row numbers for which the survival probabilities will be drawn. The default is to draw the survival probabilities for all rows.
`curemark`	if curemark = TRUE, a line will be drawn at a height that is equal to the cure rate
`conditional`	if conditional = FALSE (default), the unconditional survival probabilities will be drawn. Otherwise, the survival probabilities of uncured subjects will be drawn.
`xlab`	the label for x axis
`ylab`	the label for y axis
`ylim`	the range for y axis
`lty`	the line types for survival curves from different groups
`...`	other standard graphics parameters can be set here.

Details

This function plots survival probabilities for each row in newdata. The survival probabilities can be conditional probabilities for uncured subjects or unconditional survival probabilities, the latter will level off at estimated cure rates.

Value

a graphics window will be opened and drawn.

Author(s)

Yingwei Peng

See Also

mixcure predict.mixcure

Examples

```
data(leukaemia)
plot(predict(mixcure(Surv(time, cens) ~ transplant, ~ transplant,
data = leukaemia), newdata = leukaemia[1, ], times = 0:2000))
```

---

predict.mixcure *Prediction method for mixture cure models*

Description

This function computes the estimated survival probabilities and cure rates for given sets of covariate values using the fitted model from mixcure(). Each set of the covariates values is stored in one row of newdata. newdata must be a data frame containing all the covariates used in mixcure(). A used-supplied set of times at which the survival probabilities will be estimated must be provided in times.

Usage

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

Arguments

object	an object of mixcure
newdata	a data frame containing covariate values at which the survival and cure rate will be estimated. It should contain all the covariates that are used to build object. Prediction will be made for each row in newdata.
times	a vector of times at which the survival probabilities are estimated
...	for compatibility purpose. Not used.

Value

a list with the following components:

cure	a matrix of 2 columns and the same number of rows as newdata. The first column is uncure rates for the rows in newdata and the second column is cure rates for the rows in newdata
uncuresurv	a list with the number of components equal to the rows of newdata. Each component is a vector of the estimated survival probabilities at times for a subject if uncured
surv	similar to uncuresurv except that the survival probabilities are the unconditional survival probabilities
times	a vector of times at which the survival probabilities will be predicted

Author(s)

Yingwei Peng

See Also

mixcure

Examples

```
data(leukaemia)
predict(mixcure(Surv(time, cens) ~ transplant, ~ transplant, data = leukaemia),
newdata = leukaemia[1, ], times = 0:2000)
```

---

print.mixcure *Print method for mixture cure models*

Description

Print method for mixture cure models

Usage

```
## S3 method for class 'mixcure'
print(x, ...)
```

Arguments

x	a mixcure object
...	further arguments passed to or from other methods.

Author(s)

Yingwei Peng

See Also

mixcure

Examples

```
data(leukaemia)
z1 = mixcure(Surv(time, cens) ~ transplant, ~ transplant, data = leukaemia,
savedata = TRUE)
print(z1)
```

`residuals.mixcure` *Residuals for mixture cure models*

Description

Obtain residuals for the mixture cure models

Usage

```
## S3 method for class 'mixcure'
residuals(
  object,
  data,
  type = c("WLCH", "Cox-Snell", "M-Cox-Snell", "Martingale", "M-Martingale", "M2",
    "M3", "D2"),
  type2 = c("residuals", "partial"),
  model = c("latency", "incidence"),
  ...
)
```

Arguments

`object`	an object of mixcure
`data`	the data used to obtain mixcure object.
`type`	residuals type. WLCH is a modified residual proposed in Wileyto et al (2013). The rest types of residuals are defined in the same way as in the classic survival models.
`type2`	residual type, either "residuals" for regular residuals or "partial" for partial residuals),
`model`	if "latency" (default), residuals for latency model are generated. Otherwise, "incidence" means that residuals for incidence model are generated.
`...`	for compatibility purpose. Not used.

Details

This function computes residuals for the fitted model from mixcure().

Value

It is a list containing at least the following components:

`type`	residuals type
`residuals`	residuals of the model

Author(s)

Yingwei Peng

References

Peng, Y. and Taylor, J. M. G. Residual-based model diagnosis methods for mixture cure models. Biometrics, 73:495–505, 2017

See Also

mixcure

Examples

```
data(leukaemia)
residuals(mixcure(Surv(time, cens) ~ transplant, ~ transplant,
data = leukaemia), data = leukaemia, type = "Martingale")
```

`summary.mixcure` *Summary method for mixture cure models*

Description

Summary function will calculate bootstrap variances of the estimates in a mixcure fit if it detects that the variances do not exist. Then it will print estimates, their standard errors, z-scores and p-values from normal distribution

Usage

```
## S3 method for class 'mixcure'
summary(object, R = 100, index = 1:2, ...)
```

Arguments

object	A mixcure object
R	the number of boot samples required to calculate bootstrap variances.
index	An argument used in censboot function in boot package to specify the columns in data that store survival times and censoring indicators
...	other parameters to be passed into print function

Details

censboot in boot package is called to calculate bootstrap variances

Value

A modified mixcure object with extra components:

varboot	a censboot object
stderr	standard errors of parameters in mixcure
R	the number of bootstrap samples used

Author(s)

Yingwei Peng

See Also

mixcure

Examples

```
data(leukaemia)
# To reduce running time of this example, R is set to 2.
summary(mixcure(Surv(time, cens) ~ transplant, ~ transplant,
data = leukaemia, savedata = TRUE), R = 2)
```

Index