Package ‘YPPE’

October 12, 2022

Title  Yang and Prentice Model with Piecewise Exponential Baseline Distribution

Version  1.0.1

Description  Semiparametric modeling of lifetime data with crossing survival curves via Yang and Prentice model with piecewise exponential baseline distribution. Details about the model can be found in Demarqui and Mayrink (2019) <arXiv:1910.02406>. Model fitting carried out via likelihood-based and Bayesian approaches. The package also provides point and interval estimation for the crossing survival times.

License  GPL (>= 2)

URL  https://github.com/fndemarqui/YPPE

BugReports  https://github.com/fndemarqui/YPPE/issues

Encoding  UTF-8

LazyData  true

Biarch  true

Depends  R (>= 3.4.0), survival

Imports  methods, MASS, Formula, Rcpp (>= 0.12.0), rstan (>= 2.18.1), rstantools (>= 2.0.0)

LinkingTo  BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

SystemRequirements  GNU make

RoxygenNote  7.0.2

Suggests  testthat

NeedsCompilation  yes

Author  Fabio Demarqui [aut, cre]

Maintainer  Fabio Demarqui <fndemarqui@est.ufmg.br>

Repository  CRAN

Date/Publication  2020-01-09 20:40:03 UTC
The 'YPPE' package.

Description

Semiparametric modeling of lifetime data with crossing survival curves via Yang and Prentice model with piecewise exponential baseline distribution curves. Details about the model can be found in Demarqui and Mayrink (2019) <arXiv:1910.02406>. Model fitting carried out via likelihood-based and Bayesian approaches. The package also provides point and interval estimation for the crossing survival times.

References


**Description**

Generic S3 method `coef`

**Usage**

```r
coeff(object, ...)
```

**Arguments**

- `object`: a fitted model object
- `...`: further arguments passed to or from other methods.

**Value**

the estimated regression coefficients

---

**Description**

Estimated regression coefficients

**Usage**

```r
## S3 method for class 'yppe'
coeff(object, ...)
```

**Arguments**

- `object`: an object of the class `yppe`
- `...`: further arguments passed to or from other methods

**Value**

the estimated regression coefficients
confint

Generic S3 method confint

Description

Generic S3 method confint

Usage

confint(object, ...)

Arguments

object a fitted model object
...

... further arguments passed to or from other methods.

Value

the estimated regression coefficients

cconfint.yppe

Confidence intervals for the regression coefficients

Description

Confidence intervals for the regression coefficients

Usage

## S3 method for class 'yppe'
confint(object, level = 0.95, ...)

Arguments

object an object of the class yppe
level the confidence level required
...

... further arguments passed to or from other methods

Value

100(1-alpha) confidence intervals for the regression coefficients
**crossTime**

*Generic S3 method crossTime*

**Description**

Generic S3 method crossTime

**Usage**

```r
crossTime(object, ...)```

**Arguments**

- `object`: a fitted model object
- `...`: further arguments passed to or from other methods.

**Value**

the crossing survival time

---

**crossTime.yppe**

*Computes the crossing survival times*

**Description**

Computes the crossing survival times

**Usage**

```r
## S3 method for class 'yppe'
crossTime(object, newdata1, newdata2, conf.level = 0.95, nboot = 4000, ...)
```

**Arguments**

- `object`: an object of class yppe
- `newdata1`: a data frame containing the first set of explanatory variables
- `newdata2`: a data frame containing the second set of explanatory variables
- `conf.level`: level of the confidence/credible intervals
- `nboot`: number of bootstrap samples (default `nboot=4000`); ignored if `approach="bayes"`
- `...`: further arguments passed to or from other methods.

**Value**

the crossing survival time
Examples

```r
# ML approach:
library(YPPE)
mle <- yppe(Surv(time, status)~arm, data=ipass, approach="mle")
summary(mle)
newdata1 <- data.frame(arm=0)
newdata2 <- data.frame(arm=1)
tcross <- crossTime(mle, newdata1, newdata2)
tcross
ekm <- survfit(Surv(time, status)~arm, data=ipass)
newdata <- data.frame(arm=0:1)
St <- survfit(mle, newdata)
time <- sort(ipass$time)
plot(ekm, col=1:2)
lines(time, St[[1]])
lines(time, St[[2]], col=2)
abline(v=tcross, col="blue")

# Bayesian approach:
bayes <- yppe(Surv(time, status)~arm, data=ipass, approach="bayes")
summary(bayes)
newdata1 <- data.frame(arm=0)
newdata2 <- data.frame(arm=1)
tcross <- crossTime(bayes, newdata1, newdata2)
tcross
ekm <- survfit(Surv(time, status)~arm, data=ipass)
newdata <- data.frame(arm=0:1)
St <- survfit(bayes, newdata)
time <- sort(ipass$time)
plot(ekm, col=1:2)
lines(time, St[[1]])
lines(time, St[[2]], col=2)
abline(v=tcross, col="blue")
```

---

**gastric**

**Gastric cancer data set**

**Description**

Data set from a clinical trial conducted by the Gastrointestinal Tumor Study Group (GTSG) in 1982. The data set refers to the survival times of patients with locally nonresectable gastric cancer. Patients were either treated with chemotherapy combined with radiation or chemotherapy alone.

**Format**

A data frame with 90 rows and 3 variables:
Description

Reconstructed IPASS clinical trial data reported in Argyropoulos and Unruh (2015). Although reconstructed, this data set preserves all features exhibited in references with full access to the observations from this clinical trial. The data base is related to the period of March 2006 to April 2008. The main purpose of the study is to compare the drug gefitinib against carboplatin/paclitaxel doublet chemotherapy as first line treatment, in terms of progression free survival (in months), to be applied to selected non-small-cell lung cancer (NSCLC) patients.

Format

A data frame with 1217 rows and 3 variables:

- time: progression free survival (in months)
- status: failure indicator (1 - failure; 0 - otherwise)
- arm: (1 - gefitinib; 0 - carboplatin/paclitaxel doublet chemotherapy)

References

print.summary.yppe \hspace{1cm} Print the summary.yppe output

Description

Print the summary.yppe output

Usage

\[
\text{## S3 method for class 'summary.yppe'}
\]
\[
\text{print}(x, \ldots)
\]

Arguments

\[
x \hspace{1cm} \text{an object of the class summary.yppe.}
\]
\[
\ldots \hspace{1cm} \text{further arguments passed to or from other methods.}
\]

Value

a summary of the fitted model.

summary.yppe \hspace{1cm} Summary for the yppe model

Description

Summary for the yppe model

Usage

\[
\text{## S3 method for class 'yppe'}
\]
\[
\text{summary}(object, \ldots)
\]

Arguments

\[
\text{object} \hspace{1cm} \text{an object of the class 'yppe'.}
\]
\[
\ldots \hspace{1cm} \text{further arguments passed to or from other methods.}
\]
survfit

Generic S3 method survfit

Description

Generic S3 method survfit

Usage

survfit(object, ...)

Arguments

object

a fitted model object

...

further arguments passed to or from other methods.

Value

the crossing survival time

survfit.yppe

Survival function for the YPPE model

Description

Survival function for the YPPE model

Usage

## S3 method for class 'yppe'
survfit(object, newdata, ...)

Arguments

object

an object of the class yppe

newdata

a data frame containing the set of explanatory variables.

...

further arguments passed to or from other methods.

Value

a list containing the estimated survival probabilities.
Examples

# ML approach:
library(YPPE)
ml <- yppe(Surv(time, status)~arm, data=ipass, approach="mle")
summary(ml)
ecm <- survfit(Surv(time, status)~arm, data=ipass)
newdata <- data.frame(arm=0:1)
St <- survfit(ml, newdata)
time <- sort(ipass$time)
plot(ecm, col=1:2)
lines(time, St[[1]])
lines(time, St[[2]], col=2)

# Bayesian approach:
bayes <- yppe(Surv(time, status)~arm, data=ipass, approach="bayes")
summary(bayes)
ecm <- survfit(Surv(time, status)~arm, data=ipass)
newdata <- data.frame(arm=0:1)
St <- survfit(bayes, newdata)
time <- sort(ipass$time)
plot(ecm, col=1:2)
lines(time, St[[1]])
lines(time, St[[2]], col=2)

---

**timeGrid**

**Time grid**

Description

Time grid

Usage

timeGrid(time, status, n_int = NULL)

Arguments

time Vector of failure times
status Vector of failure indicators
n_int Optional. Number of intervals. If NULL, the number of intervals is set to be equal to the number of distinct observed failure times.

Value

Time grid.
**vcov**

**Generic S3 method vcov**

**Description**

Generic S3 method vcov

**Usage**

vcov(object, ...)

**Arguments**

- object: a fitted model object
- ...: further arguments passed to or from other methods.

**Value**

the variance-covariance matrix associated the regression coefficients.

---

**vcov.yppe**

**Covariance of the regression coefficients**

**Description**

Covariance of the regression coefficients

**Usage**

## S3 method for class 'yppe'
vcov(object, ...)

**Arguments**

- object: an object of the class yppe
- ...: further arguments passed to or from other methods.

**Value**

the variance-covariance matrix associated with the regression coefficients.
yppe  

Fits the Yang and Prentice model with baseline distribution modelled by the piecewise exponential distribution.

Description

Fits the Yang and Prentice model with baseline distribution modelled by the piecewise exponential distribution.

Usage

yppe(
  formula,  
data,  
n_int = NULL,  
rho = NULL,  
tau = NULL,  
hessian = TRUE,  
approach = c("mle", "bayes"),  
hyper_parms = list(h1_gamma = 0, h2_gamma = 4, mu_psi = 0, sigma_psi = 4, mu_phi = 0, sigma_phi = 4, mu_beta = 0, sigma_beta = 4),  
...  
)

Arguments

formula an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which yppe is called.
n_int number of intervals of the PE distribution. If NULL, default value (square root of n) is used.
rho the time grid of the PE distribution. If NULL, the function timeGrid is used to compute rho.
tau the maximum time of follow-up. If NULL, tau = max(time), where time is the vector of observed survival times.
hessian logical; If TRUE (default), the hessian matrix is returned when approach="mle".
approach approach to be used to fit the model (mle: maximum likelihood; bayes: Bayesian approach).
hyper_parms a list containing the hyper-parameters of the prior distributions (when approach = "bayes"). If not specified, default values are used.
... Arguments passed to either 'rstan::optimizing' or 'rstan::sampling'.

...
yppe returns an object of class "yppe" containing the fitted model.

Examples

# ML approach:
library(YPPE)
mle <- yppe(Surv(time, status)~arm, data=ipass, approach="mle")
summary(mle)

# Bayesian approach:
bayes <- yppe(Surv(time, status)~arm, data=ipass, approach="bayes")
summary(bayes)
Index

* datasets
  gastric, 6
  ipass, 7

  coef, 3
  coef.yppe, 3
  confint, 4
  confint.yppe, 4
  crossTime, 5
  crossTime.yppe, 5

  gastric, 6
  ipass, 7

  print.summary.yppe, 8

  summary.yppe, 8
  survfit, 9
  survfit.yppe, 9

  timeGrid, 10

  vcov, 11
  vcov.yppe, 11

  YPPE (YPPE-package), 2
  yppe, 12
  YPPE-package, 2