## Package ‘spsurv’

October 14, 2022

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Bernstein Polynomial Based Semiparametric Survival Analysis</td>
</tr>
<tr>
<td>Version</td>
<td>1.0.0</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Renato Panaro <a href="mailto:renatovp@ime.usp.br">renatovp@ime.usp.br</a></td>
</tr>
<tr>
<td>Description</td>
<td>A set of reliable routines to ease semiparametric survival regression modeling based on Bernstein polynomials. 'spsurv' includes proportional hazards, proportional odds and accelerated failure time frameworks for right-censored data. RV Panaro (2020) [arXiv:2003.10548].</td>
</tr>
<tr>
<td>License</td>
<td>GPL-3</td>
</tr>
<tr>
<td>Biarch</td>
<td>true</td>
</tr>
<tr>
<td>Depends</td>
<td>R (&gt;= 3.5.0), survival (&gt;= 2.44-1.1), loo (&gt;= 2.1.0), coda (&gt;= 0.19-3), MASS (&gt;= 7.3-51.4)</td>
</tr>
<tr>
<td>Imports</td>
<td>methods, Rcpp (&gt;= 0.12.0), rstan (&gt;= 2.18.1), rstantools (&gt;= 1.5.1)</td>
</tr>
<tr>
<td>LinkingTo</td>
<td>BH (&gt;= 1.66.0), Rcpp (&gt;= 0.12.0), RcppEigen (&gt;= 0.3.3.3.0), rstan (&gt;= 2.18.1), StanHeaders (&gt;= 2.18.0)</td>
</tr>
<tr>
<td>Suggests</td>
<td>knitr, devtools, roxygen2, testthat, KMsurv</td>
</tr>
<tr>
<td>Encoding</td>
<td>UTF-8</td>
</tr>
<tr>
<td>LazyData</td>
<td>true</td>
</tr>
<tr>
<td>SystemRequirements</td>
<td>GNU make</td>
</tr>
<tr>
<td>BugReports</td>
<td><a href="https://github.com/rvpanaro/spsurv/issues">https://github.com/rvpanaro/spsurv/issues</a></td>
</tr>
<tr>
<td>VignetteBuilder</td>
<td>knitr</td>
</tr>
<tr>
<td>RoxygenNote</td>
<td>7.0.2</td>
</tr>
<tr>
<td>NeedsCompilation</td>
<td>yes</td>
</tr>
<tr>
<td>Author</td>
<td>Renato Panaro [aut, cre] (<a href="https://orcid.org/0000-0002-1903-2091">https://orcid.org/0000-0002-1903-2091</a>), <a href="https://rvpanaro.github.io/">https://rvpanaro.github.io/</a>), Fábio Demarqui [aut] (<a href="https://orcid.org/0000-0001-9236-1986">https://orcid.org/0000-0001-9236-1986</a>), Vinicius Mayrink [aut] (<a href="https://orcid.org/0000-0002-5683-8326">https://orcid.org/0000-0002-5683-8326</a>)</td>
</tr>
<tr>
<td>Repository</td>
<td>CRAN</td>
</tr>
<tr>
<td>Date/Publication</td>
<td>2020-03-31 10:00:02 UTC</td>
</tr>
</tbody>
</table>
The 'spsurv' package.

Description

A set of flexible routines to allow semiparametric survival regression modeling based on Bernstein polynomial, including Bernstein based proportional hazards model (BPPH), Bernstein polynomial based proportional odds model (BPPO) and Bernstein based accelerated failure time model (BPAFT) for right-censored data.
Details

spbp fits semi-parametric models for time-to-event survival data. Non-informative right-censoring assumption is available. Any user-defined Bernstein polynomial can be user-defined using an arbitrary degree, i.e. highest basis polynomials order.

The framework takes advantage of fully likelihood methods since the polynomial parameters are used to estimate the baseline functions. Even so, this is said to be semi-parametric since this approach does not rely on any distribution. Unlike the Cox model, the BP based models provide smooth hazard and survival curve estimates.

Value

none

Author(s)

Renato Valladares Panaro (renatovp@ime.usp.br).

References


bp.basis

Bernstein basis polynomials calculations

Description

Bernstein basis polynomials calculations

Usage

bp.basis(time, degree, tau = max(time))

Arguments

  time a vector of times.
  degree Bernstein polynomial degree
  tau must be greater than times maximum value observed.
Value

A list containing matrices b and B corresponding BP basis and corresponding tau value used to compute them.

Description

Fits the BPAFT model to time-to-event data.

Usage

bpaft(formula, degree, data, approach = c("mle", "bayes"), ...)

Arguments

- **formula**: a Surv object with time to event observations, right censoring status and explanatory terms.
- **degree**: Bernstein polynomial degree.
- **data**: a data.frame object.
- **approach**: Bayesian or maximum likelihood estimation methods, default is approach = "mle".
- **...**: further arguments passed to or from other methods

Value

An object of class 'spbp'.

See Also

spbp, bpph and bppo for other BP based models.

Examples

```r
library("spsurv")
data("veteran")

fit <- bpaft(Surv(time, status) ~ karno + celltype, data = veteran)

summary(fit)
```
**bpph**  
*Bernstein Polynomial Based Proportional Hazards Model*

**Description**

Fits the BPPH model to time-to-event data.

**Usage**

```r
bpph(formula, degree, data, approach = c("mle", "bayes"), ...)
```

**Arguments**

- `formula`: a Surv object with time to event observations, right censoring status and explanatory terms.
- `degree`: Bernstein polynomial degree.
- `data`: a data.frame object.
- `approach`: Bayesian or maximum likelihood estimation methods, default is approach = "mle".
- `...`: further arguments passed to or from other methods

**Value**

An object of class `spbp`.

**See Also**

`spbp`, `bppo` and `bpaft` for other BP based models.

**Examples**

```r
library("spsurv")
data("veteran")

fit <- bpph(Surv(time, status) ~ karno + factor(celltype),
            data = veteran)

summary(fit)
```
Bernstein Polynomial Based Proportional Odds Model

Description
Fits the BPPO model to time-to-event data.

Usage
bppo(formula, degree, data, approach = c("mle", "bayes"), ...)

Arguments
- formula: a Surv object with time-to-event observations, right censoring status and explanatory terms.
- degree: Bernstein polynomial degree.
- data: a data.frame object.
- approach: Bayesian or maximum likelihood estimation methods, default is approach = "mle".
- ... further arguments passed to or from other methods

Value
An object of class `spbp`.

See Also
- spbp, bpph and bpaft for other BP based models.

Examples
library("spsurv")
data("veteran")

fit <- bppo(Surv(time, status) ~ karno + celltype, data = veteran)
summary(fit)
**Description**

Generic S3 method `coef`

**Usage**

`coef(sbp, ...)`

**Arguments**

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

**Value**

the estimated regression coefficients

---

**Description**

Estimated regression coefficients

**Usage**

```
## S3 method for class 'spbp'
coef(sbp, ...)
```

**Arguments**

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

**Value**

the estimated regression coefficients
### confint

*Generic S3 method confint*

**Description**

Generic S3 method confint

**Usage**

```r
confint(spbp, ...) 
```

**Arguments**

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

**Value**

the estimated regression coefficients

### confint.spbp

*Confidence intervals for the regression coefficients*

**Description**

Confidence intervals for the regression coefficients

**Usage**

```r
## S3 method for class 'spbp'
confint(spbp, level = 0.95, ...) 
```

**Arguments**

- `spbpm`: an object of the class spbp
- `level`: the confidence level required
- `...`: further arguments passed to or from other methods

**Value**

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

*Generic S3 method extract*

**Description**

Generic S3 method extract

**Usage**

extract(sbp, ...)

**Arguments**

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

**Value**

extract the MCMC chain values of a Bayesian fit.

---

**extract.spbp**

*Extract method for fitted spbp models*

**Description**

Extract samples from a fitted spbp model.

**Usage**

```r
## S3 method for class 'spbp'
extract(sbp, pars = c("beta", "gamma"), ...)
```

**Arguments**

- `spbp` an object of class `spbp` result of a spbp fit.
- `pars` parameters to be selected.
- `...` arguments inherent from extract.

**Value**

see extract.

**See Also**

- spbp, stan_dens.spbp, traceplot.spbp
library("spsurv")
data("veteran")

fit <- bpph(Surv(time, status) ~ karno + factor(celltype),
            data = veteran)

extract(fit)

itsamp

Inverse Transform Sampling To Generate Time-to-event Data From
Parametric Models

Description
Random survival times generation for the weibull or log-logistic distributions with parameters
'scale' and 'shape'.

Usage
itsamp(
n, 
  beta = c(2, -1),
  event_scale = 10,
  censor_scale = 4,
  features = data.frame(x1 = rnorm(n, 0), x2 = rnorm(n, 0)),
  shape = 2,
  model = c("ph", "po", "aft"),
  dist = c("weibull", "llogis"),
  censor = TRUE
)

Arguments

n
  integer; sample size

beta
  vector of regression coefficients

event_scale, censor_scale
  event and censoring scale parameters

features
  matrix of features (columns)

shape
  event and censoring distribution shape

model
  either "ph" (default) or "aft" for weibull and "po" or "aft" for log-logistic distribution

dist
  "weibull" or "llogis"

censor
  logical; if TRUE, censoring is required, that is mean(status) > 0
Details

`sim_surv` returns weibull (log-logistic) randomly generated survival times. According to Collett (2003), the accelerated failure time model encompasses a wide variety of parametric models, including weibull and log-logistic models.

Value

data.frame of `ncol(x) + 2` columns in which the survival times are the response variable denoted by 'y', 'status' indicates failure (0 = failure) and the features are appended to the next columns.

See Also

`spbp`

Examples

```r
rows <- 200
categorical <- rbinom(rows, size = 3, prob = .5)
x <- data.frame(numerical = rnorm(rows),
cat0 = as.numeric(categorical == 0),
cat1 = as.numeric(categorical == 1),
cat2 = as.numeric(categorical == 2),
cat3 = as.numeric(categorical == 3))

newdata <- itsamp(n = rows, beta = c(1, -2, .5, .1, 1),
features = x, model = 'ph', dist = 'weibull')
```

---

**Description**

Calculate the posterior mode

**Usage**

```r
mode(ext)
```

**Arguments**

- `ext` rstan extracted sample.

**Value**

A vector containing the posterior mode of each sample.
Description

Model.matrix of a fitted `spbp` model.

Usage

```r
## S3 method for class 'spbp'
model.matrix(
  object,
  data = eval(object$call$data, envir = parent.frame()),
  ...)
```

Arguments

- `object` an object of class `spbp`, see `spbp`.
- `data` data.frame object.
- `...` arguments inherent from `model.matrix`.

Value

The explanatory variables matrix.

See Also

`spbp`, `model.matrix`

Examples

```r
library("spsurv")
data("veteran")

fit <- bpph(Surv(time, status) ~ karno + factor(celltype),
             data = veteran)

model.matrix(fit)
```
print.spbp

Bernstein Polynomial Based Regression Object Print

Description
Bernstein Polynomial Based Regression Object Print

Usage
## S3 method for class 'spbp'
print(
  x,
  digits = max(getOption("digits") - 4, 3),
  signif.stars = getOption("show.signif.stars"),
  ...  
)

Arguments
x an object of class spbp
digits number of digits to display
signif.stars see getOption
...
further arguments passed to or from other methods

Value
none

print.summary.bpaft.bayes

Bernstein Polynomial Based Regression Object Summary BPAFT Bayes

Description
Bernstein Polynomial Based Regression Object Summary BPAFT Bayes

Usage
## S3 method for class 'summary.bpaft.bayes'
print(...)

Arguments
... further arguments passed to or from other methods
print.summary.bpph.bayes

Value

none

print.summary.bpaft.mle

Bernstein Polynomial Based Regression Object Summary BPAFT MLE

Description

Bernstein Polynomial Based Regression Object Summary BPAFT MLE

Usage

## S3 method for class 'summary.bpaft.mle'
print(...)

Arguments

... further arguments passed to or from other methods

Value

none

print.summary.bpph.bayes

Bernstein Polynomial Based Regression Object Summary BPPH Bayes

Description

Bernstein Polynomial Based Regression Object Summary BPPH Bayes

Usage

## S3 method for class 'summary.bpph.bayes'
print(...)

Arguments

... further arguments passed to or from other methods

Value

none
Bernstein Polynomial Based Regression Object Summary BPPH MLE

Description
Bernstein Polynomial Based Regression Object Summary BPPH MLE

Usage
## S3 method for class 'summary.bpph.mle'
print(...)

Arguments
...

Value
none

Bernstein Polynomial Based Regression Object Summary BPPO Bayes

Description
Bernstein Polynomial Based Regression Object Summary BPPO Bayes

Usage
## S3 method for class 'summary.bppo.bayes'
print(...)

Arguments
...

Value
none
print.summary.bppo.mle

Bernstein Polynomial Based Regression Object BPPO MLE

Description

Bernstein Polynomial Based Regression Object BPPO MLE

Usage

## S3 method for class 'summary.bppo.mle'
print(...)

Arguments

... further arguments passed to or from other methods

Value

none

print.summary.spbp.bayes

Bernstein Polynomial Based Regression Object Summary Bayes

Description

Bernstein Polynomial Based Regression Object Summary Bayes

Usage

## S3 method for class 'summary.spbp.bayes'
print(x, digits = max(getOption("digits") - 4, 3), ...)

Arguments

x a summary.spbp.bayes object
digits number of digits to display.
... further arguments passed to or from other methods

Value

none
Bernstein Polynomial Based Regression Object Summary MLE

Description
Bernstein Polynomial Based Regression Object Summary MLE

Usage
## S3 method for class 'summary.spbp.mle'
print(
  x,
  digits = max(getOption("digits") - 4, 3),
  signif.stars = getOption("show.signif.stars"),
  ...
)

Arguments
- **x**: a summary.spbp.mle object
- **digits**: number of digits to display.
- **signif.stars**: see `getOption`
- **...**: further arguments passed to or from other methods

Value
none

residuals.spbp
BP based models residuals.

Description
Residuals for a fitted `spbp` model.

Usage
## S3 method for class 'spbp'
residuals(object, type = c("cox-snell"), ...)

Arguments
- **object**: an object of class `spbp` result of a `spbp` fit.
- **type**: type of residuals, default is "cox-snell"
- **...**: further arguments passed to or from other methods
See Also

spbp.

Examples

library("spsurv")
data("veteran")

fit <- bpph(Surv(time, status) ~ karno + factor(celltype),
            data = veteran)

residuals(fit)

spbp: The BP Based Survival Analysis Function

Description

Semiparametric Survival Analysis Using Bernstein Polynomial

Usage

spbp(formula, ...)

Arguments

formula a Surv object with time to event, status and explanatory terms.

... Arguments passed to ‘rstan::sampling’ (e.g. iter, chains) or ‘rstan::optimizing’.

Details

Fits Bernstein Polynomial based Proportional regression to survival data.

Value

An object of class ‘spbp’.

See Also

spbp.default
spbp.default, bpph, bppo, bpaft, https://mc-stan.org/users/documentation/
Examples

library("spsurv")
data("veteran") # imports from survival package

fit_mle <- spbp(Surv(time, status) ~ karno + factor(celltype),
data = veteran, model = "po")
summary(fit_mle)

fit_bayes <- spbp(Surv(time, status) ~ karno + factor(celltype),
data = veteran, model = "po", approach = "bayes",
cores = 1, iter = 300, chains = 1,
priors = list(beta = c("normal(0,4)"),
gamma = "lognormal(0,4)"))

summary(fit_bayes)

spbp.default

spbp: The BP Based Semiparametric Survival Analysis Function

Description

spbp: The BP Based Semiparametric Survival Analysis Function

Usage

## Default S3 method:

spbp(
  formula,
  degree,
  data,
  approach = c("mle", "bayes"),
  model = c("ph", "po", "aft"),
priors = list(beta = c("normal(0,4)"), gamma = "lognormal(0,10)"),
scale = TRUE,
cores = parallel::detectCores(),
...)

Arguments

formula a Surv object with time to event, status and explanatory terms
degree Bernstein Polynomial degree
data a data.frame object
approach Bayesian or Maximum Likelihood estimation methods, default is approach = "bayes"
model
Proportional Hazards or Proportional Odds BP based regression, default is model = "ph"
priors
prior settings for the Bayesian approach; ‘normal’ or ‘cauchy’ for beta; ‘gamma’, ‘inv_gamma’ or ‘lognormal’ for gamma (BP coefficients)
scale
logical; indicates whether to center and scale the data
cores
number of core threads to use
...
further arguments passed to or from other methods

Value
An object of class spbp

stan_dens
Generic S3 method extract

Description
Generic S3 method extract

Usage
stan_dens(spbp, ...)

Arguments
spbp    a fitted model object
...

Value
the density plot of a MCMC chain.

stan_dens.spbp
Density plots method for fitted spbp models

Description
Posterior density of samples from a fitted spbp model.

Usage
## S3 method for class 'spbp'
stan_dens(spbp, pars = c("beta", "gamma"), ...)

...
Arguments

spbp  the result of a spbp fit.
pars  parameters to be selected.
...  arguments inherent from stan_dens.

Value

see stan_dens.

See Also

spbp, traceplot.spbp, extract.spbp

Examples

library("spsurv")
data("veteran")

fit <- bpph(Surv(time, status) ~ karno + factor(celltype),
            data = veteran)

stan_dens(fit)
survivor

Generic S3 method vcov

Description

Generic S3 method vcov

Usage

survivor(spbp, ...)

Arguments

spbp an object of class spbp
... further arguments passed to or from other methods

Value

estimates survival for each dataset individual (line). Spbp Object Observed Survival

See Also

spbp, itsamp

---

traceplot

Generic S3 method traceplot

Description

Generic S3 method traceplot

Usage

traceplot(spbp, ...)

Arguments

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

Value

the traceplot of a MCMC chain.
traceplot.spbp

Traceplot method for fitted spbp models

Description

Traceplot of a Bayesian fit spbp.

Usage

## S3 method for class 'spbp'
traceplot(spbp, pars = c("beta", "gamma"), ...)

Arguments

spbp an object of class 'spbp' result of a spbp fit.
pars parameters to be selected.
... arguments inherent from traceplot.

Value

see traceplot.

See Also

spbp, stan_dens.spbp, extract.spbp

Examples

library("spsurv")
data("veteran")

fit <- bpph(Surv(time, status) ~ karno + factor(celltype),
            data = veteran)

traceplot(fit)
vcov

*Generic S3 method vcov*

**Description**

Generic S3 method vcov

**Usage**

vcov(spbp, ...)

**Arguments**

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

**Value**

the variance-covariance matrix associated the regression coefficients.

---

vcov.spbp

*Covariance of the regression coefficients*

**Description**

Covariance of the regression coefficients

**Usage**

```r
## S3 method for class 'spbp'
vcov(spbp, ...)
```

**Arguments**

- **spb**
  - an object of the class spbp
- **...**
  - further arguments passed to or from other methods.

**Value**

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

bp.basis, 3
bpaft, 4, 5, 6, 18
bpph, 4, 5, 6, 18
bppo, 4, 5, 6, 18
coef, 7
gooff.spbp, 7
confint, 8
confint.spbp, 8
extract, 9, 9
extract.spbp, 9, 21, 23
getOption, 13, 17
itsamp, 10, 22
mode, 11
model.matrix, 12
model.matrix.spbp, 12
print.spbp, 13
print.summary.bpaft.bayes, 13
print.summary.bpaft.mle, 14
print.summarybpph.bayes, 14
print.summary.bpph.mle, 15
print.summary.bppo.bayes, 15
print.summary.bppo.mle, 16
print.summary.spbp.bayes, 16
print.summary.spbp.mle, 17
residuals.spbp, 17
spbp, 3–6, 9, 11, 12, 17, 18, 18, 20–23
spbp.default, 18, 19
spsurv(spsurv-package), 2
spsurv-package, 2
stan_dens, 20, 21
stan_dens.spbp, 9, 20, 23
summary.spbp, 21
survivor, 22

traceplot, 22, 23
traceplot.spbp, 9, 21, 23
vcov, 24
vcov.spbp, 24