

# Package: spsurv (via r-universe)

July 5, 2026

**Type** Package

**Title** Bernstein Polynomial Based Semiparametric Survival Analysis

**Version** 1.0.2

**Author** Renato Panaro [aut, cre, cph], Fábio Demarqui [ctb, ths],  
Vinicius Mayrink [ths]

**Maintainer** Renato Panaro <rvpanaro@gmail.com>

**Description** Semiparametric survival analysis based on Bernstein polynomials. 'spsurv' includes proportional hazards, proportional odds and accelerated failure time frameworks for right-censored data. RV Panaro (2020) <doi:10.48550/arXiv.2003.10548>.

**License** GPL-3

**Biarch** true

**Depends** R (>= 3.5.0), survival (>= 2.44-1.1), loo (>= 2.9.0), coda (>= 0.19-3), MASS (>= 7.3-51.4)

**Imports** methods, Rcpp (>= 0.12.0), rstan (>= 2.18.1), magrittr, viridis, mnormt

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

**Suggests** devtools, roxygen2, testthat (>= 2.1.0), KMSurv, covr, rmarkdown

**Encoding** UTF-8

**LazyData** true

**SystemRequirements** GNU make

**BugReports** <https://github.com/rvpanaro/spsurv/issues>

**RoxygenNote** 7.3.3

**URL** <https://github.com/rvpanaro/spsurv>,  
<https://rvpanaro.github.io/spsurv>

**Config/pak/sysreqs** make

**Repository** <https://rvpanaro.r-universe.dev>

**Date/Publication** 2026-03-08 22:57:50 UTC

**RemoteUrl** <https://github.com/rvpanaro/spsurv>

**RemoteRef** HEAD

**RemoteSha** 476e32cc1fd1d4e4552d6ae28c9c8a78b14845ff

## Contents

spsurv-package	2
bp.basis	3
bpaft	4
bpph	5
bppo	6
coef.spbp	7
confint.spbp	7
credint	8
credint.spbp	8
model.matrix.spbp	9
plot.spbp	9
print.spbp	10
print.summary.bpaft.bayes	11
print.summary.bpaft.mle	12
print.summary.bpph.bayes	12
print.summary.bpph.mle	13
print.summary.bppo.bayes	13
print.summary.bppo.mle	14
print.summary.spbp.bayes	14
print.summary.spbp.mle	15
pw.basis	15
residuals.spbp	16
spbp	17
spbp.default	18
summary.spbp	19
survfit.spbp	19
vcov.spbp	20
<b>Index</b>	<b>21</b>

---

spsurv-package	<i>The 'spsurv' package.</i>
----------------	------------------------------

---

## 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**

`sbbp` 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.

`_PACKAGE`

**Value**

none

**Author(s)**

rvpanaro@gmail.com

**References**

Panaro R.V. (2020). `spsurv`: An R package for semi-parametric survival analysis. arXiv preprint arXiv:2003.10548.

Demarqui, F. N., & Mayrink, V. D. (2019). A fully likelihood-based approach to model survival data with crossing survival curves. arXiv preprint arXiv:1910.02406.

Demarqui, F. N., Mayrink, V. D., & Ghosh, S. K. (2019). An Unified Semiparametric Approach to Model Lifetime Data with Crossing Survival Curves. arXiv preprint arXiv:1910.04475.

Osman, M., & Ghosh, S. K. (2012). Nonparametric regression models for right-censored data using Bernstein polynomials. *Computational Statistics & Data Analysis*, 56(3), 559-573.

Lorentz, G. G. (1953). Bernstein polynomials. American Mathematical Society.

---

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  $g$  and  $G$  corresponding BP basis and corresponding tau value used to compute them.

---

 bpaft

*Bernstein Polynomial Based Accelerated Failure Time Model*


---

**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](#), [bpbh](#) and [bppo](#) for other BP based models.

**Examples**

```
library("spsurv")
data("veteran", package = "survival")

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

summary(fit)
```

**Description**

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

**Usage**

```
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**

```
library("spsurv")
data("veteran", package = "survival")

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

summary(fit)
```

---

**bppo***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", package = "survival")

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

summary(fit)
```

---

coef.spbp	<i>Estimated regression coefficients</i>
-----------	--

---

**Description**

Estimated regression coefficients

**Usage**

```
## S3 method for class 'spbp'
coef(object, summary = c("mean", "median", "mode"), ...)
```

**Arguments**

object	an object of the class spbp
summary	posterior summary if method = "bayes" in x
...	further arguments passed to or from other methods

**Value**

the estimated regression coefficients

---

confint.spbp	<i>Confidence intervals for the regression coefficients</i>
--------------	---

---

**Description**

Confidence intervals for the regression coefficients

**Usage**

```
## S3 method for class 'spbp'
confint(object, parm = names(coef(object)), level = 0.95, ...)
```

**Arguments**

object	a fitted model object.
parm	a specification of which parameters are to be given confidence intervals, either a vector of numbers or a vector of names. If missing, all parameters are considered.
level	the confidence level required.
...	further arguments passed to parent method

**Value**

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

---

credint	<i>Generic S3 method credint</i>
---------	----------------------------------

---

**Description**

Generic S3 method credint

**Usage**

```
credint(x, ...)
```

**Arguments**

x	a fitted model object
...	further arguments passed to parent method

**Value**

the estimated regression coefficients

---

credint.spbp	<i>Confidence intervals for the regression coefficients</i>
--------------	---

---

**Description**

Confidence intervals for the regression coefficients

**Usage**

```
## S3 method for class 'spbp'
credint(x, prob = 0.95, type = c("HPD", "Equal-Tailed"), ...)
```

**Arguments**

x	an object of the class x.
prob	the probability level required.
type	interval type.
...	further arguments passed to or from other methods.

**Value**

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

---

model.matrix.spbp	<i>Model.matrix method for fitted spbp models</i>
-------------------	---

---

### Description

Model.matrix of a fitted [spbp](#) model.

### Usage

```
## S3 method for class 'spbp'  
model.matrix(object, ...)
```

### Arguments

object	an object of class 'spbp', see <a href="#">spbp</a> .
...	arguments passed to parent method.

### Value

The model matrix.

### See Also

[spbp](#), [model.matrix](#)

### Examples

```
library("spsurv")  
data("veteran", package = "survival")  
  
fit <- bpph(Surv(time, status) ~ karno + factor(celltype),  
  data = veteran  
)  
  
model.matrix(fit)
```

---

plot.spbp	<i>BP based models plot.</i>
-----------	------------------------------

---

### Description

Plot for a fitted [spbp](#) model.

**Usage**

```
## S3 method for class 'spbp'
plot(
  x,
  main,
  graph = c("baseline", "basis"),
  cumulative = F,
  frame = F,
  lwd = 3,
  ...
)
```

**Arguments**

x	an object of class 'spbp' result of a <a href="#">spbp</a> fit.
main	graph title
graph	type of polynomial graph, default is "basis"
cumulative	TRUE for odds and cumulative hazard
frame	graphical parameter; default is FALSE
lwd	graphical parameter; default is 3
...	further arguments passed to or from other methods

**See Also**

[spbp](#).

**Examples**

```
library("spsurv")
data("veteran", package = "survival")

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

---

print.spbp

*Bernstein Polynomial Based Regression Object Print*


---

**Description**

Bernstein Polynomial Based Regression Object Print

**Usage**

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

**Arguments**

x	an object of class spbp.
bp.param	print BP parameters.
digits	number of digits to display.
signif.stars	see <a href="#">getOption</a> .
...	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
-----	---

**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
```

---

print.summary.bpph.mle

*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**

... further arguments passed to or from other methods

**Value**

none

---

print.summary.bppo.bayes

*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**

... further arguments passed to or from other methods

**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

---

```
print.summary.spbp.mle
```

*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 <a href="#">getOption</a>
...	further arguments passed to or from other methods

### Value

none

---

```
pw.basis
```

*Power basis polynomials calculations*

---

### Description

Power basis polynomials calculations

### Usage

```
pw.basis(degree)
```

### Arguments

degree	Bernstein polynomial degree
--------	-----------------------------

**Value**

A list containing matrices  $g$  and  $G$  corresponding BP basis and corresponding tau value used to compute them.

---

residuals.spbp      *BP based models residuals.*

---

**Description**

Residuals for a fitted [spbp](#) model.

**Usage**

```
## S3 method for class 'spbp'  
residuals(object, type = c("martingale", "deviance", "coobject-snell"), ...)
```

**Arguments**

object	an object of class 'spbp' result of a <a href="#">spbp</a> fit.
type	type of residuals, default is "cox-snell"
...	arguments passed to parent method.

**See Also**

[spbp](#), [spbp](#).

**Examples**

```
library("spsurv")  
data("veteran", package = "survival")  
  
fit <- bpph(Surv(time, status) ~ karno + factor(celltype),  
  data = veteran  
)  
  
residuals(fit)
```

---

spbp

*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", package = "survival")

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(0,5)"),
    gamma = "halfnormal(0,5)"
  )
)
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 = c("lognormal(0,4)"), frailty =
    c("gamma(0.01,0.01)")),
  cores = min(parallel::detectCores() - 1, 4),
  scale = TRUE,
  verbose = FALSE,
  chains = 4,
  ...
)
```

## 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; 'lognormal' or 'loglogistic' for gamma (BP coefficients)
cores	number of core threads to use
scale	logical; indicates whether to center and scale the data
verbose	verbose passed to stan
chains	number of chains passed to stan
...	further arguments passed to or from other methods

## Value

An object of class spbp

---

summary.spbp

*Bernstein Polynomial Based Regression Object Summary*


---

**Description**

Bernstein Polynomial Based Regression Object Summary

**Usage**

```
## S3 method for class 'spbp'
summary(object, interval = 0.95, ...)
```

**Arguments**

object	an object of class spbp
interval	interval coverage (confidence or credibility)
...	further arguments passed to or from other methods

**Value**

An object of class analogous to for e.g. 'summary.bppo.bayes'.

---

survfit.spbp

*BP-based model survival curves*


---

**Description**

Compute survival curves for a fitted [spbp](#) model.

**Usage**

```
## S3 method for class 'spbp'
survfit(
  formula,
  newdata,
  times,
  se.fit = TRUE,
  interval = 0.95,
  type = c("log", "log-log", "plain"),
  ...
)
```

**Arguments**

formula	An object of class "spbp" returned by <a href="#">spbp</a> .
newdata	Optional data frame used to obtain survival curves for specific covariate values.
times	Optional numeric vector of time points at which to return estimates.
se.fit	Logical; if TRUE, compute standard errors.
interval	Confidence level for intervals (e.g. 0.95).
type	Character; confidence interval transformation. One of "log", "log-log", or "plain".
...	Further arguments (currently ignored or reserved for future use).

**Value**

An object of class "survfit".

**See Also**

[spbp](#), [survfit](#).

**Examples**

```
library(spsurv)
data(veteran, package = "survival")
fit <- bpph(Surv(time, status) ~ karno + factor(celltype), data = veteran)
survfit(fit)
```

---

vcov.spbp

*Covariance of the regression coefficients*


---

**Description**

Uses block-wise inversion of the negative Hessian, with a clear split between the regression coefficients (beta) and the Bernstein polynomial coefficients (gamma).

**Usage**

```
## S3 method for class 'spbp'
vcov(object, bp.param = FALSE, ...)
```

**Arguments**

object	an object of the class spbp
bp.param	return Bernstein Polynomial variance.
...	arguments passed to parent method.

**Value**

the variance-covariance matrix associated with the regression coefficients.

# Index

bp.basis, 3  
bpaft, 4, 5, 6, 17  
bpph, 4, 5, 6, 17  
bppo, 4, 5, 6, 17  
  
coef.spbp, 7  
confint.spbp, 7  
credint, 8  
credint.spbp, 8  
  
getOption, 11, 15  
  
model.matrix, 9  
model.matrix.spbp, 9  
  
plot.spbp, 9  
print.spbp, 10  
print.summary.bpaft.bayes, 11  
print.summary.bpaft.mle, 12  
print.summary.bpph.bayes, 12  
print.summary.bpph.mle, 13  
print.summary.bppo.bayes, 13  
print.summary.bppo.mle, 14  
print.summary.spbp.bayes, 14  
print.summary.spbp.mle, 15  
pw.basis, 15  
  
residuals.spbp, 16  
  
spbp, 3–6, 9, 10, 16, 17, 19, 20  
spbp.default, 17, 18  
spsurv (spsurv-package), 2  
spsurv-package, 2  
summary.spbp, 19  
survfit, 20  
survfit.spbp, 19  
  
vcov.spbp, 20