

# Package ‘TwoWaySurvival’

February 14, 2012

**Title** Additiv Two-Way Hazards Modelling of Right Censored Survival Data

**Version** 2.2

**Author** Pavel Khomski <pkhomski@web.de>

**Description** The package offers a fitting of smooth varying coefficients in an additiv two-way hazard model. A one-way hazard model can also be fitted with supplementary functions. Nonparametric penalized spline (p-spline) fitting is proposed. In the two-way case two alternativ models can be analyzed: In the first one a non periodic calendar time, like the calendar year, is considered as a second time scale (additionaly to the survival time), and as a spline basis truncated polynomial functions are chosen. In the second model a periodic time scale, like a season of year, is additionally considered, and as a spline basis the b-splines are selected. In the one-way case the user can choose between these two alternative penalized bases.

**Maintainer** Pavel Khomski <pkhomski@web.de>

**Depends** splines

**License** GPL (>= 2)

**Repository** CRAN

**Date/Publication** 2007-11-27 08:24:57

## R topics documented:

OneWaySurv . . . . .	2
OneWaySurvfitControl . . . . .	3
OneWaySurvfitCreate . . . . .	4
OneWaySurvfitObject . . . . .	6
plot.OneWaySurvfit . . . . .	7
plot.TwoWaySurvfit . . . . .	8
print.OneWaySurvfit . . . . .	9

print.TwoWaySurvfit . . . . .	10
seasonal . . . . .	11
summary.OneWaySurvfit . . . . .	11
summary.TwoWaySurvfit . . . . .	12
TwoWaySurv . . . . .	13
TwoWaySurvfitControl . . . . .	14
TwoWaySurvfitCreate . . . . .	16
TwoWaySurvfitObject . . . . .	18
unemployed . . . . .	19

<b>Index</b>	<b>20</b>
--------------	-----------

---

OneWaySurv	<i>Creates a One-Way Survival Object</i>
------------	--

---

### Description

Creates an object of class 'OneWaySurv' for using as a response in the model formula in the call to the 'OneWaySurvfitCreate' function.

### Usage

```
OneWaySurv(surv.time, status)
```

### Arguments

surv.time	the time to follow up for right censored data.
status	the (0,1)-indicator, 1 for event and 0 for censored observation.

### Details

The function handles only right censored data. All arguments must be numeric vectors. NA's can be supplied but will not be supported by the function 'OneWaySurvfitCreate' in the further analysis.

### Value

An object of class 'OneWaySurv'.

### Author(s)

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

### References

Kauermann G.(2005): Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

**See Also**[OneWaySurvfitCreate](#)**Examples**

```

surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.OneWaySurv.object<-OneWaySurv(surv.time=surv.time,status=status)

```

---

OneWaySurvfitControl    *Control Values for Fitting of two-way Hazards*

---

**Description**

The values supplied in the call of 'OneWaySurvfitCreate' function will replace the defaults, and a list with all possible arguments is returned. The returned list is used as the 'control' argument to the 'OneWaySurvfitCreate' function.

**Usage**

```

OneWaySurvfitControl(niter.epoch = 100, niter.penalty = 2, tol.epoch.theta = 1e-08, tol.epoch.variance
                    tol.penalty = 1e-08, print.epoch = FALSE, print.penalty=FALSE, print.log.lik=FALSE,
                    print.estimate=FALSE, method="fix", number.int=60, ...)

```

**Arguments**

niter.epoch	maximum number of iterations of the outer loop in the backfitting optimization algorithm.
niter.penalty	maximum number of iterations of the inner loop in the backfitting optimization algorithm.
tol.epoch.theta	tolerance for the convergence criterion for the fixed and random parameters.
tol.epoch.variance	tolerance for the convergence criterion for the penalty values w.r.t. outer loop.
tol.penalty	tolerance for the convergence criterion for the penalty values w.r.t. inner loop.
print.epoch	logical value for printing results from each outer loop.
print.penalty	logical value for printing results from each inner loop.
print.log.lik	logical value for printing the marginal log-likelihood in each iteration.
print.estimate	logical value for printing estimates of the fixed parameters and penalty from each iteration.
method	character string specifying optimization method for smoothing parameter; currently can be only "fix" meaning optimization via fix point iteration.
number.int	a specified number of the integration points for approximation of the cumulative hazard function

... other parameters which can only be 'num.knots.t' for the number of spline knots for survival time, as well as starting value 'start.penalty.t' for penalized terms for the random parts of the model.

### Details

The defaults or user specified values are applied as the 'control' argument in the call of the 'OneWaySurvfitCreate' function. It can be an (empty) list object or a call to the 'OneWaySurvfitControl' function itself, whether or not with supplied arguments to be changed from their default values. If not supplied, the value for 'num.knots.t' will be determined internally. The value of 'number.int' should be chosen moderately; this because of generation of artificial poisson distributed data, for each observation respectively. (See supplemented paper describing the generation mechanism.)

### Value

a list with components for each of the possible arguments.

### Author(s)

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

### References

Kauerman G. (2005). Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

### See Also

[OneWaySurvfitCreate](#)

---

OneWaySurvfitCreate    *Additiv One-Way Hazards Modelling with Varying Coefficients*

---

### Description

This function is for the fitting of models with hazard depending on a survival time only. Other time scales are not considered (compare for this with 'TwoWaySurvfitCreate' function). The effects of covariates are modelled as varying coefficients. For flexible smoothing of the effects whether a basis of truncated polynomial splines or the B-splines can be chosen.

### Usage

```
OneWaySurvfitCreate(formula=formula, data=data, na.action=na.fail, control=control, basis=basis)
```

**Arguments**

formula	a symbolic description of the model to be fit. The details of model specification are given below.
data	an optional data frame containing variables in the model. If not supplied the handling of missing values will be accordingly to the value of 'na.action'.
na.action	an optional parameter for handling with missing data.
control	an optional list argument with control parameters needed for optimization routine. If not specified, default values from 'OneWaySurvfitControl' will be taken; otherwise the values, which are defined by user, will overwrite the defaults.
basis	character string needed for specifying the function basis, which will be chosen for estimating the varying coefficients. Can only be one of 'trunc' or 'bspline'

**Details**

A formula argument can be specified in different ways. The response or left part of the formula can be either an object of class 'OneWaySurv' or it can be a call to the 'OneWaySurv' function with its arguments specified in it. The right part may be left unspecified (without '~') or it can be of '~1' type. In this case no covariates are supplied to the model, hence only baseline hazard will be adopted. In the case with covariates these must be supplied explicitly through their names in a standard way. In all cases the 'formula' argument must be supplied.

If factor variables are to be considered, their reference categories should be specified before applied for optimization.

A 'control' argument can be absent, in this case the defaults from the 'OneWaySurvfitControl' will be taken, or it can be either a list with some or all components of it specified by their names with according values, or it can be a call to the 'OneWaySurvfitControl' function with parameters specified in it.

The most of computation and optimization is implemented in the "internal" functions 'TruncatedOneway' or 'BsplineOneway', which are called from the body of 'OneWaySurvfitCreate' in depending on the bases, which are chosen for the fitting routine. The code for both routines is well commented and can be clearly understood following the supplemented paper.

The computation procedure can take considerable time, depending on data supplied. All evaluations are written in R-code, so no external program code like C or Fortran has been used. We advice initially to set the number of iterations not to be too large, and eventually increase it if needed.

**Value**

An object of class 'OneWaySurvfit'. It has methods for 'print', 'summary' and 'plot'.

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauermann G.(2005). Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

**See Also**

[OneWaySurv](#), [OneWaySurvfitControl](#), [OneWaySurvfitObject](#)

**Examples**

```
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.OneWaySurv.object<-OneWaySurv(surv.time=surv.time,status=status)
my.OneWaySurvfit.object<-OneWaySurvfitCreate(formula=my.OneWaySurv.object, na.action=na.fail,
      control = list(niter.epoch=50,print.epoch=FALSE,method="fix"),basis="bspline")
x1<-rnorm(20)
x2<-rt(20,2)
my.OneWaySurvfit.object<-OneWaySurvfitCreate(formula = my.OneWaySurv.object~x1+x2,data=data.frame(x1,x2),
      na.action = na.fail,control = OneWaySurvfitControl(niter.epoch=50,print.epoch=FALSE,method="fix"))
```

---

OneWaySurvfitObject     *Fitted one-dimensional Object*

---

**Description**

An object of class 'OneWaySurvfit' returned as a list by the 'OneWaySurvfitCreate' function and representing a fitted one-way hazard model with varying coefficients. It has methods for generic functions print, plot and summary.

**Value**

fix.coef	named vector of not penalized parameters of the model. It has the value 'NULL' if B-spline bases are chosen for modelling.
random.coef	named vector of penalized parameters of the model.
penalty	named vector of penalty values for the random parts of the model.
var.fix	estimated variances of not penalized parameters of the model. It has the value 'NULL' if B-spline bases are chosen for modelling.
var.random	estimated variances of the predicted values of (penalized) random components of the model.
log.lik.margin.start	initial value of the marginal log-likelihood.
log.lik.margin	the value of the marginal log-likelihood at optimum.
df	estimated degrees of freedom for the parts of the model.
df.total	total or summed estimated degrees of freedom.
niter.epoch	number of outer loops in optimization routine.
varying.frame	data frame with named columns, according to the names of the varying coefficients. It is needed for plotting of the smooth components.

deviation.frame	data frame with named columns, according to the names of the varying coefficients. It is needed for plotting of the confidence bands of the smooth components.
grid.frame	data frame with just one column, for survival time, needed for plotting issues.
p	number of covariates; for factors including their categories (excluding reference category).
factor.names	covariate names; for factors the names of categories (excluding reference category).

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauerman G. (2005). Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

**See Also**

[OneWaySurv](#), [OneWaySurvfitCreate](#)

---

plot.OneWaySurvfit     *Plots the Smoothed Varying Coefficients*

---

**Description**

'plot.OneWaySurvfit' is a proposed function for plotting of smooth components of the one-dimensional model, one plot for each baseline and covariate (or factor level) effect, for survival time.

**Usage**

```
## S3 method for class 'OneWaySurvfit'
plot(x,...)
```

**Arguments**

x                    object of class 'OneWaySurvfit'  
 ...                    additional plot parameters

**Details**

All plots will be made in the device, which is specified by the user. One can use the object components for producing his own plots.

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauermann G. (2005). Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

**See Also**

[print.OneWaySurvfit](#), [summary.OneWaySurvfit](#)

**Examples**

```
pdf(file="myplot.pdf",h=12,w=12)
par(mfrow=c(1,2))
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,1,0,0,1,0,1)
my.OneWaySurv.object<-OneWaySurv(surv.time=surv.time,status=status)
x<-OneWaySurvfitCreate(my.OneWaySurv.object~1)
plot.OneWaySurvfit(x)
dev.off()
```

---

plot.TwoWaySurvfit      *Plots the Smoothed Varying Coefficients*

---

**Description**

'plot.TwoWaySurvfit' is a proposed function for plotting of smooth components of the model, one plot for each baseline and covariate (or factor level) effect, for survival time and birth time accordingly.

**Usage**

```
## S3 method for class 'TwoWaySurvfit'
plot(x,...)
```

**Arguments**

```
x                    object of class 'TwoWaySurvfit'
...                   additional plot parameters
```

**Details**

All plots will be made in the device, which is specified by the user. One can use the object components for producing his own plots.

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauermann G. and Khomski P. (2006). Additiv two way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956

**See Also**

[print.TwoWaySurvfit](#), [summary.TwoWaySurvfit](#)

**Examples**

```
pdf(file="myplot.pdf",h=12,w=12)
par(mfrow=c(1,2))
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
birth.time<-c(1,11,23,4,23,5,7,12,5,12,23,12,15,21,4,6,1,4,13,11)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.TwoWaySurv.object<-TwoWaySurv(surv.time=surv.time,birth.time=birth.time,status=status)
x<-TwoWaySurvfitCreate(my.TwoWaySurv.object~1,component="non.periodic")
plot.TwoWaySurvfit(x)
dev.off()
```

---

print.OneWaySurvfit    *Prints an Object of class 'OneWaySurvfit'*

---

**Description**

Prints estimates of fixed parameters of smoothing varying coefficients as well as penalty values for their random parts.

**Usage**

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

**Arguments**

x                    object of class 'OneWaySurvfit'  
...                   additional parameters for print

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauerman G. (2005). Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

**See Also**

[summary.OneWaySurvfit](#), [plot.OneWaySurvfit](#)

**Examples**

```
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.OneWaySurv.object<-OneWaySurv(surv.time=surv.time,status=status)
x<-OneWaySurvfitCreate(my.OneWaySurv.object)
print.OneWaySurvfit(x)
```

---

```
print.TwoWaySurvfit    Prints an Object of class 'TwoWaySurvfit'
```

---

**Description**

Prints estimates of fixed parameters of smoothing varying coefficients as well as penalty values for their random parts.

**Usage**

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

**Arguments**

```
x          object of class 'TwoWaySurvfit'
...        additional parameters for print
```

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauermann G. and Khomski P. (2006). Additiv two way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956.

**See Also**

[summary.TwoWaySurvfit](#), [plot.TwoWaySurvfit](#)

**Examples**

```

surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
birth.time<-c(1,11,23,4,23,5,7,12,5,12,23,12,15,21,4,6,1,4,13,11)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.TwoWaySurv.object<-TwoWaySurv(surv.time=surv.time,birth.time=birth.time,status=status)
x<-TwoWaySurvfitCreate(my.TwoWaySurv.object,component="non.periodic")
print.TwoWaySurvfit(x)

```

---

seasonal

*Simulated data with periodic component*


---

**Description**

This data give the 1000 simulated observations with survival time and a periodic (or seasonal) component time scale. One additional covariate is simulated too.

**Usage**

```
data(seasonal)
```

**Format**

A data frame containing 1000 sampled observations. Description of columns:

- birthSimulated seasonal time of entry in the study or starting point, like season of birth in a year.
- survSimulated survival time.
- statusSimulated censoring indicator in the usual notation for right censored data.
- covariateSimulated covariate.

---

summary.OneWaySurvfit *Objects Summaries*


---

**Description**

prints detailed results from the fitting of the one-way hazards modelling.

**Usage**

```

## S3 method for class 'OneWaySurvfit'
summary(object,...)

```

**Arguments**

```

object      an object of class 'OneWaySurvfit'
...         additional arguments

```

**Value**

the printed values are estimates of fixed and random components of varying coefficients with their estimated variances as well as penalty values and marginal log-likelihoods at start point and at optimum.

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauerman G. (2005). Penalised Spline Fitting in Multivariable Survival Models with Varying Coefficients *Computational Statistics and Data Analysis*, 49, 169-186.

**See Also**

[print.OneWaySurvfit](#), [plot.OneWaySurvfit](#)

**Examples**

```
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.OneWaySurv.object<-OneWaySurv(surv.time=surv.time,status=status)
x<-OneWaySurvfitCreate(my.OneWaySurv.object)
summary.OneWaySurvfit(x)
```

---

summary.TwoWaySurvfit *Objects Summaries*

---

**Description**

prints detailed results from the fitting of the two-way hazards modelling.

**Usage**

```
## S3 method for class 'TwoWaySurvfit'
summary(object,...)
```

**Arguments**

object	an object of class 'TwoWaySurvfit'
...	additional arguments

**Value**

the printed values are estimates of fixed and random components of varying coefficients with their estimated variances as well as penalty values and marginal log-likelihoods at start point and at optimum.

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauermann G. and Khomski P. (2006). Additiv two way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956.

**See Also**

[print.TwoWaySurvfit](#), [plot.TwoWaySurvfit](#)

**Examples**

```
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
birth.time<-c(1,11,23,4,23,5,7,12,5,12,23,12,15,21,4,6,1,4,13,11)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.TwoWaySurv.object<-TwoWaySurv(surv.time=surv.time,birth.time=birth.time,status=status)
x<-TwoWaySurvfitCreate(my.TwoWaySurv.object,component="non.periodic")
summary.TwoWaySurvfit(x)
```

---

TwoWaySurv

*Creates a Two-Way Survival Object*

---

**Description**

Creates an object of class 'TwoWaySurv' for using as a response in the model formula in the call to the 'TwoWaySurvfitCreate' function.

**Usage**

```
TwoWaySurv(surv.time, birth.time, status)
```

**Arguments**

surv.time	the time to follow up for right censored data.
birth.time	the (calendar) time to enter in the study.
status	the (0,1)-indicator, 1 for event and 0 for censored observation.

**Details**

The function handles only right censored data. All arguments must be numeric vectors. NA's can be supplied but will not be supported by the function 'TwoWaySurvfitCreate' in the further analysis.

**Value**

An object of class 'TwoWaySurv'.

**Author(s)**

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

**References**

Kauermann G. and Khomski P. (2006): Additiv two-way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956.

**See Also**

[TwoWaySurvfitCreate](#)

**Examples**

```
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
birth.time<-c(1,11,23,4,23,5,7,12,5,12,23,12,15,21,4,6,1,4,13,11)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,0,0,1,0,1)
my.TwoWaySurv.object<-TwoWaySurv(surv.time=surv.time,birth.time=birth.time,status=status)
```

---

TwoWaySurvfitControl    *Control Values for Fitting of two-way Hazards*

---

**Description**

The values supplied in the call of 'TwoWaySurvfitCreate' function will replace the defaults, and a list with all possible arguments is returned. The returned list is used as the 'control' argument to the 'TwoWaySurvfitCreate' function.

**Usage**

```
TwoWaySurvfitControl(niter.epoch = 100, niter.penalty = 2, tol.epoch.theta = 1e-08, tol.epoch.variance
  tol.penalty = 1e-08, print.epoch = FALSE, print.penalty=FALSE, print.log.lik=FALSE,
  print.estimate=FALSE, method="fix", number.int=60, ...)
```

**Arguments**

niter.epoch	maximum number of iterations of the outer loop in the backfitting optimization algorithm.
niter.penalty	maximum number of iterations of the inner loop in the backfitting optimization algorithm.
tol.epoch.theta	tolerance for the convergence criterion for the fixed and random parameters.
tol.epoch.variance	tolerance for the convergence criterion for the penalty values w.r.t. outer loop.
tol.penalty	tolerance for the convergence criterion for the penalty values w.r.t. inner loop.
print.epoch	logical value for printing results from each outer loop.

<code>print.penalty</code>	logical value for printing results from each inner loop.
<code>print.log.lik</code>	logical value for printing the marginal log-likelihoods in each iteration.
<code>print.estimate</code>	logical value for printing estimates of the fixed parameters and penalty from each iteration.
<code>method</code>	specified optimization method for smoothing parameter; can be either "fix" meaning optimization via fix point iteration, or "NR" meaning Newton-Raphson optimization. We advice to set <code>method="fix"</code> because evaluations then seem to go faster.
<code>number.int</code>	a specified number of the integration points for approximation of the cumulative hazard function
<code>...</code>	other parameters which can only be <code>'num.knots.t'</code> and <code>'num.knots.b'</code> for the number of spline knots for survival time and birth time accordingly, as well as starting values <code>'start.penalty.t'</code> and <code>'start.penalty.b'</code> for penalized terms for the random parts of the model. The latter should be applied together, otherwise both of them will be initially set to 1 internally.

### Details

The defaults or user specified values are applied as the `'control'` argument in the call of the `'TwoWaySurvfitCreate'` function. It can be an (empty) list object or a call to the `'TwoWaySurvfitControl'` function itself, whether or not with supplied arguments to be changed from their default values. If not supplied, the values for `'num.knots.t'` or `'num.knots.b'` will be determined internally. The value of `'number.int'` should be chosen moderately; this because of generation of artificial poisson distributed data, for each observation respectively. (See supplemented paper describing the generation mechanism.)

### Value

a list with components for each of the possible arguments.

### Author(s)

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

### References

Kauerman G. and Khomski P. (2006). Additiv two way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956.

### See Also

[TwoWaySurvfitCreate](#)

---

TwoWaySurvfitCreate     *Additiv Two-Way Hazards Modelling with Varying Coefficients*

---

### Description

This function is for the fitting of models with hazard depending on both survival time and calendar (here birth or entry in the study) time. The effects of covariates are modelled as varying coefficients, additive in both times. For flexible smoothing of the effects truncated polynomial splines are employed.

### Usage

```
TwoWaySurvfitCreate(formula=formula, data=data, na.action=na.fail, control=control, component=component)
```

### Arguments

formula	a symbolic description of the model to be fit. The details of model specification are given below.
data	an optional data frame containing variables in the model. If not supplied the handling of missing values will be accordingly to the value of 'na.action'.
na.action	an optional parameter for handling with missing data.
control	an optional list argument with control parameters needed for optimization routine. If not specified, default values from 'TwoWaySurvfitControl' will be taken; otherwise the values, which are defined by user, will overwrite the defaults.
component	character string parameter which specifies the model to be estimated. If component='non.periodic' then a model with nonperiodic calendar effects (like a year) will be estimated. If component='periodic' then a model with a periodic component (like a seasonal one) will be estimated.

### Details

A formula argument can be specified in different ways. The response or left part of the formula can be either an object of class 'TwoWaySurv' or it can be a call to the 'TwoWaySurv' function with its arguments specified in it. The right part may be left unspecified (without '~') or it can be of '~1' type. In this case no covariates are supplied to the model, hence only baseline hazard will be adopted. In the case with covariates these must be supplied explicitly through their names in a standard way. In all cases the 'formula' argument must be supplied.

If factor variables are to be considered, their reference categories should be specified before applied for optimization.

A 'control' argument can be absent, in this case the defaults from the 'TwoWaySurvfitControl' will be taken, or it can be either a list with some or all components of it specified by their names with according values, or it can be a call to the 'TwoWaySurvfitControl' function with parameters specified in it.

The most of computation and optimization is implemented in the "internal" functions 'BackfitTruncated' or 'BackfitBspline', which are called from the body of 'TwoWaySurvfitCreate' in depending

on what kind of a second time scale (nonperiodic or periodic) is considered. For the nonperiodic component the basis of truncated polynomials are employed. For the periodic component the estimation routine is done via B-splines. This choice makes it also necessary to do additional transformation of the B-spline basis, which is called wrapping, to smoothly join the ends of splines. The code for both routines is well commented and can be clearly understood following the supplemented paper.

The computation procedure can take considerable time, depending on data supplied. All evaluations are written in R-code, so no external program code like C or Fortran has been used. We advise initially to set the number of iterations not to be too large, and eventually increase it if needed.

### Value

An object of class 'TwoWaySurvfit'. It has methods for 'print', 'summary' and 'plot'.

### Author(s)

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

### References

Kauermann G. and Khomski P. (2006). Additive two way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956.

### See Also

[TwoWaySurv](#), [TwoWaySurvfitControl](#), [TwoWaySurvfitObject](#)

### Examples

```
surv.time<-c(1,2,4,3,5,6,1,6,2,3,4,7,5,1,6,7,10,2,10,5)
birth.time<-c(1,11,23,4,23,5,7,12,5,12,23,12,15,21,4,6,1,4,13,11)
status<-c(1,1,1,0,1,1,1,0,0,1,1,1,1,1,1,1,0,0,1,0,1)
my.TwoWaySurv.object<-TwoWaySurv(surv.time=surv.time,birth.time=birth.time,status=status)
my.TwoWaySurvfit.object<-TwoWaySurvfitCreate(formula=my.TwoWaySurv.object, na.action=na.fail,
      control = list(niter.epoch=50,print.epoch=FALSE,method="fix"),
      component="non.periodic")
x1<-rnorm(20)
x2<-rt(20,2)
my.TwoWaySurvfit.object<-TwoWaySurvfitCreate(formula = my.TwoWaySurv.object~x1+x2,data=data.frame(x1,x2),
      na.action = na.fail,control = TwoWaySurvfitControl(niter.epoch=50,print.epoch=FALSE,method="fix"),
      component="non.periodic")
```

---

TwoWaySurvfitObject    *Fitted Object*

---

### Description

An object of class 'TwoWaySurvfit' returned as a list by the 'TwoWaySurvfitCreate' function and representing a fitted additive two-way hazard model with varying coefficients. It has methods for generic functions print, plot and summary.

### Value

<code>fix.coef</code>	named vector of fixed parameters, beta, of the model.
<code>random.coef</code>	named vector of random parameters, u, of the model.
<code>penalty</code>	named vector of penalty values for the random parts of the model.
<code>var.fix</code>	estimated variances of the fixed components, beta, of the model.
<code>var.random</code>	estimated variances of the predicted values of random components, u, of the model.
<code>log.lik.margin.start</code>	initial value of the marginal log-likelihoods for survival time (t-) and calendar time (b-) directions.
<code>log.lik.margin</code>	the value of the marginal log-likelihoods for t- and b- directions at optimum.
<code>df</code>	estimated degrees of freedom for the parts of the model.
<code>df.total</code>	total or summed estimated degrees of freedom.
<code>niter.epoch</code>	number of outer loops in optimization routine.
<code>varying.frame</code>	data frame with named columns, according to the names of the varying coefficients, for each t- and b- direction. It is needed for plotting of the smooth components.
<code>deviation.frame</code>	data frame with named columns, according to the names of the varying coefficients, for each t- and b- direction. It is needed for plotting of the confidence bands of the smooth components.
<code>grid.frame</code>	data frame with two columns, the first for survival time and the second for the birth time, needed for plotting issues.
<code>p</code>	number of covariates; for factors including their categories (excluding reference category).
<code>factor.names</code>	covariate names; for factors the names of categories (excluding reference category).

### Author(s)

Pavel Khomski <pkhomski@wiwi.uni-bielefeld.de>

## References

Kauermann G. and Khomski P. (2006). Additiv two way hazards model with varying coefficients. *Computational Statistics and Data Analysis*, 51, 1944-1956.

## See Also

[TwoWaySurv](#), [TwoWaySurvfitCreate](#)

---

unemployed

*Sample of german unemployed*

---

## Description

This data give the time spent in the unemployment spell and the calendar time of entry in that spell for a sample of german unemployed individuals. The full data have been resived from the *Deutsches Institut fuer Wirtschaftsforschung* and were analysed in the larger study. The results can be found in the supplemented paper.

## Usage

```
data(unemployed)
```

## Format

A data frame containing 300 sampled observations. Description of columns:

- birthCalendar time of entry in the unemployment spell, measured in months, ranging from the year 1983 till 2000.
- survTime spent in the unemployment spell, measured in months. The spells which lasted more than 36 months have been truncated on 36 months and denoted as sensed.
- statusCensoring indicator in the usual notation for right censored data.
- nationNationality of the unemployed, german vs. others.
- genderGender of the person.
- ageAge has been categorized in young (till 25), middleage and old (over 50).
- trainingWhether an individual received a professional training.
- universityWhether an individual has a university degree.

## Source

Socio Economic Panel (SOEP), s. [www.diw.de/deutsch/sop](http://www.diw.de/deutsch/sop).

## References

Kauermann G. and Khomski P. (2006). Additiv two way hazards model with varying coefficients, in press.

# Index

- \*Topic **datasets**
    - seasonal, [11](#)
    - unemployed, [19](#)
  - \*Topic **hplot**
    - plot.OneWaySurvfit, [7](#)
    - plot.TwoWaySurvfit, [8](#)
  - \*Topic **list**
    - OneWaySurvfitControl, [3](#)
    - TwoWaySurvfitControl, [14](#)
  - \*Topic **misc**
    - OneWaySurv, [2](#)
    - OneWaySurvfitObject, [6](#)
    - TwoWaySurv, [13](#)
    - TwoWaySurvfitObject, [18](#)
  - \*Topic **nonparametric**
    - OneWaySurvfitCreate, [4](#)
    - TwoWaySurvfitCreate, [16](#)
  - \*Topic **print**
    - print.OneWaySurvfit, [9](#)
    - print.TwoWaySurvfit, [10](#)
    - summary.OneWaySurvfit, [11](#)
    - summary.TwoWaySurvfit, [12](#)
  - \*Topic **smooth**
    - OneWaySurvfitCreate, [4](#)
    - TwoWaySurvfitCreate, [16](#)
  - \*Topic **survival**
    - OneWaySurvfitCreate, [4](#)
    - TwoWaySurvfitCreate, [16](#)
- OneWaySurv, [2](#), [6](#), [7](#)  
OneWaySurvfitControl, [3](#), [6](#)  
OneWaySurvfitCreate, [3](#), [4](#), [4](#), [7](#)  
OneWaySurvfitObject, [6](#), [6](#)
- plot.OneWaySurvfit, [7](#), [10](#), [12](#)  
plot.TwoWaySurvfit, [8](#), [10](#), [13](#)  
print.OneWaySurvfit, [8](#), [9](#), [12](#)  
print.TwoWaySurvfit, [9](#), [10](#), [13](#)
- seasonal, [11](#)
- summary.OneWaySurvfit, [8](#), [10](#), [11](#)  
summary.TwoWaySurvfit, [9](#), [10](#), [12](#)
- TwoWaySurv, [13](#), [17](#), [19](#)  
TwoWaySurvfitControl, [14](#), [17](#)  
TwoWaySurvfitCreate, [14](#), [15](#), [16](#), [19](#)  
TwoWaySurvfitObject, [17](#), [18](#)
- unemployed, [19](#)