

# Package ‘contsurvplot’

August 16, 2022

**Title** Visualize the Effect of a Continuous Variable on a Time-to-Event Outcome

**Version** 0.1.0

**Maintainer** Robin Denz <robin.denz@rub.de>

**Description** Graphically display the causal effect of a continuous variable on a time-to-event outcome using multiple different types of plots based on g-computation. Those functions include, among others, survival area plots, survival contour plots, survival quantile plots and 3D surface plots. Due to the use of g-computation, all plot allow confounder-adjustment naturally. For details, see Robin Denz, Nina Timmesfeld (2022) <[arXiv:2208.04644v1](#)>.

**License** GPL (>= 3)

**URL** <https://github.com/RobinDenz1/contsurvplot>,  
<https://robindenz1.github.io/contsurvplot/>

**BugReports** <https://github.com/RobinDenz1/contsurvplot/issues>

**Imports** ggplot2 (>= 3.0.0), dplyr, rlang, riskRegression, foreach

**Suggests** survival, pammttools, ganimate, transformr, plotly, reshape2,  
doParallel, knitr, rmarkdown, testthat (>= 3.0.0), vdiff (>= 1.0.0), covr

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Contact** <robin.denz@rub.de>

**Encoding** UTF-8

**RoxygenNote** 7.2.0

**NeedsCompilation** no

**Author** Robin Denz [aut, cre]

**Repository** CRAN

**Date/Publication** 2022-08-16 09:20:11 UTC

## R topics documented:

contsurvplot-package . . . . .	2
curve_cont . . . . .	3
plot_surv_3Dsurface . . . . .	6
plot_surv_animated . . . . .	9
plot_surv_area . . . . .	13
plot_surv_at_t . . . . .	17
plot_surv_contour . . . . .	20
plot_surv_heatmap . . . . .	23
plot_surv_lines . . . . .	27
plot_surv_quantiles . . . . .	30
plot_surv_rmst . . . . .	33
plot_surv_rmtl . . . . .	36
<b>Index</b>	<b>40</b>

---

contsurvplot-package	<i>Visualize the Effect of a Continuous Variable on a Time-To-Event Outcome</i>
----------------------	---

---

## Description

### *What is this package about?*

This package provides different plotting routines to visualize the causal effect of a continuous variable on a time-to-event outcome using a previously fit model and g-computation. Unlike simpler alternatives, such as plotting survival curves for some categories, these plots always correspond to the results obtained by the time-to-event model and give an accurate depiction of the causal effect, if all assumptions are met.

### *What features are included in this package?*

The package includes 10 different plot functions, most based on the **ggplot2** package. Those 10 plotting routines include *value specific survival curves*, *landmark survival probability plots*, *survival time quantile plots*, *survival probability heatmaps*, and *survival area plots*, among others. A description and comparison of these plots can be found in the article associated with this R-package (Denz & Timmesfeld 2022).

### *What does a typical workflow using this package look like?*

All the user has to do is fit a time-to-event model, such as the cox-model, including the continuous variable of interest (and possibly confounders) and plug it into one of the plot functions included in this package. Many different kind of models are supported. See [curve\\_cont](#) for more details.

### *What type of plot should I use?*

There is no general answer to this question, but we would usually suggest using a plot method that is able to visualize the causal survival probability both as a function of time and as a function of the continuous variable. The [plot\\_surv\\_area](#), [plot\\_surv\\_heatmap](#) and [plot\\_surv\\_contour](#) functions do just that. More discussion about this topic can be found in the vignette and the associated paper.

***Where can I get more information?***

The documentation pages contain a lot of information, relevant examples and literature references. Additional examples can be found in the vignette of this package, which can be accessed using `vignette(topic="introduction", package="contsurvplot")`. We are also currently working on an article focused on the two novel plotting methods `plot_surv_area` and `plot_surv_contour`.

***I want to suggest a new feature / I want to report a bug. Where can I do this?***

Bug reports, suggestions and feature requests are highly welcome. Please file an issue on the official github page (<https://github.com/RobinDenz1/contsurvplot>) or contact the author directly using the supplied e-mail address.

**Author(s)**

Robin Denz (robin.denz@rub.de)

**References**

Robin Denz, Nina Timmesfeld (2022). Visualizing the Causal Effect of a Continuous Variable on a Time-To-Event Outcome. arXiv:2208.04644v1

James Robins. A New Approach to Causal Inference in Mortality Studies with a Sustained Exposure Period: Application to Control of the Healthy Worker Survivor Effect. *Mathematical Modelling* (1986) 7, pages 1393-1512.

---

curve_cont	<i>Calculate Causal Survival or Failure Probabilities for Levels of a Continuous Variable</i>
------------	---

---

**Description**

This function can be utilized to calculate survival curves or cumulative incidence functions (CIF) for specific values of a continuous covariate.

**Usage**

```
curve_cont(data, variable, model, horizon,
           times, group=NULL, cause=1, cif=FALSE, n_cores=1,
           na.action=options()$na.action, ...)
```

**Arguments**

data	A <code>data.frame</code> containing all required variables.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the <code>data.frame</code> that is supplied to the <code>data</code> argument.
model	A model describing the time-to-event process (such as a <code>coxph</code> model). Needs to include <code>variable</code> as an independent variable. It also has to have an associated <code>predictRisk</code> method. See <code>?predictRisk</code> for more details.

horizon	A numeric vector containing a range of values of variable for which the survival curves should be calculated.
times	A numeric vector containing points in time at which the survival probabilities should be calculated.
group	An optional single character string specifying a factor variable in data. When used, the regression standardization is performed conditional on this factor variable, meaning that one estimate is returned for each level of the factor variable. See details for a better description. Set to NULL (default) to use no grouping variable.
cause	The cause of interest. In standard survival data with only one event type, this should be kept at 1. For data with multiple failure types, this argument should be specified. In addition, the <code>cif</code> argument should be set to TRUE in those cases.
cif	Whether to calculate the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. In those cases, this argument should always be set to TRUE.
n_cores	The number of processor cores to use when performing the calculations. If <code>n_cores=1</code> (default), single threaded processing is used. If <code>n_cores &gt; 1</code> the <b>foreach</b> package and the <b>doParallel</b> package are used to run the calculations on <code>n_cores</code> in parallel. This might speed up the runtime considerably when it is initially slow.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
...	Further arguments passed to <code>predictRisk</code> .

## Details

This function uses *Direct Standardization*, also known as the *Corrected Group Prognosis* method, *G-Computation* or *G-Formula*, to calculate the causal survival probability or CIF for values of a continuous variable. The resulting estimates can be interpreted as the average survival probability (or failure probability) that would have been observed if the value of `variable` had been set to a specific value for all individuals in data.

If the model supplied to `model` is correct, the resulting estimates will be unbiased. By including relevant confounders as independent covariates in the `model`, the resulting estimates are adjusted for the confounders as well. This function is used internally in all plot functions included in this R-package and generally does not need to be called directly by the user. It can however be used to get specific values or as a basis to create custom plots not included in this package.

It relies on the `predictRisk` function from the **riskRegression** package to create the covariate and time specific estimates of the probabilities. If there are multiple failure types in the status variable of the time-to-event outcome, this should be directly modeled using a suitable model, such as a Cause-Specific-Cox model or a Fine & Gray model. In this case, only the cause-specific CIF can be estimated and `cif=TRUE` should always be used accordingly.

If `group` is not NULL, the whole estimation will be performed once for each of the factor levels of `group`. This means that in addition to fixing the level of the continuous covariate at a certain value

for the entire dataset, the grouping variable is fixed as well. The results can be interpreted similarly to the usual estimate, only that they are also conditional on the level of group used. All plots that support this argument will be faceted according to the group variable. To obtain useful estimates, it is crucial to include the group variable in the model.

### Value

Returns a data.frame containing the columns time (the point in time), surv (the estimated survival probability) and cont (the specific value of variable used). If group was used, it includes the additional group column, specifying the level of the grouping variable.

### Author(s)

Robin Denz

### References

Brice Ozenne, Anne Lyngholm Sorensen, Thomas Scheike, Christian Torp-Pedersen and Thomas Alexander Gerds. riskRegression: Predicting the Risk of an Event using Cox Regression Models. The R Journal (2017) 9:2, pages 440-460.

I-Ming Chang, Rebecca Gelman, and Marcello Pagano. Corrected Group Prognostic Curves and Summary Statistics. Journal of Chronic Diseases (1982) 35, pages 669-674

James Robins. A New Approach to Causal Inference in Mortality Studies with a Sustained Exposure Period: Application to Control of the Healthy Worker Survivor Effect. Mathematical Modelling (1986) 7, pages 1393-1512.

### See Also

[predictRisk](#)

### Examples

```
library(condsurvplot)
library(riskRegression)
library(survival)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# estimate survival probability at some points in time, for
# a range of age values
plotdata <- curve_cont(data=nafld1,
                       variable="age",
```

```

model=model,
horizon=c(50, 60, 70, 80),
times=c(1000, 2000, 3000, 4000))

# estimate cumulative incidences instead
plotdata <- curve_cont(data=nafl1,
  variable="age",
  model=model,
  horizon=c(50, 60, 70, 80),
  times=c(1000, 2000, 3000, 4000),
  cif=TRUE)

```

---

plot\_surv\_3Dsurface *Plot the Survival Curve or CIF Dependent on a Continuous Variable as a 3D Surface*

---

### Description

Using a previously fit time-to-event model, this function plots the survival curve as a function of a continuous variable using a 3D surface representation. This function can produce interactive plots using the **plotly** package, or a static image using the **persp** function of the **graphics** package.

### Usage

```

plot_surv_3Dsurface(time, status, variable, data, model,
  cif=FALSE, na.action=options()$na.action,
  horizon=NULL, fixed_t=NULL, max_t=Inf,
  interactive=FALSE,
  xlab="Time", ylab="Survival Probability",
  zlab=variable, ticktype="detailed",
  theta=120, phi=20, col="green", shade=0.5,
  ...)

```

### Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as an coxph model). Needs to include variable as an independent variable. It also has to have an associated <b>predictRisk</b> method. See <code>?predictRisk</code> for more details.

<code>cif</code>	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
<code>na.action</code>	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
<code>horizon</code>	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 40 equally spaced steps.
<code>fixed_t</code>	A numeric vector containing points in time at which the survival probabilities should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the survival probability is estimated at every point in time at which an event occurred.
<code>max_t</code>	A number indicating the latest survival time which is to be plotted.
<code>interactive</code>	Whether to draw the 3D surface as a static plot ( <code>interactive=FALSE</code> , the default) or as an interactive plot. When <code>interactive=TRUE</code> is used, this function relies on the <b>plotly</b> package. In this case, the only aesthetic arguments that still work are the <code>xlab</code> , <code>ylab</code> and <code>zlab</code> arguments. Everything else has to be set manually using the functionality of the <b>plotly</b> package directly.
<code>xlab</code>	A character string used as the x-axis label of the plot, representing the survival time.
<code>ylab</code>	A character string used as the y-axis label of the plot, representing the survival probability or CIF.
<code>zlab</code>	A character string used as the z-axis of the plot, representing the continuous covariate of interest.
<code>ticktype</code>	A single character: "simple" draws just an arrow parallel to the axis to indicate direction of increase; "detailed" draws normal ticks as per 2D plots. Passed to the <code>ticktype</code> argument in the <code>persp</code> function. Ignored if <code>interactive=TRUE</code> .
<code>theta</code>	Angles defining the viewing direction. <code>theta</code> gives the azimuthal direction and <code>phi</code> the colatitude. Passed to the <code>theta</code> argument in the <code>persp</code> function. Ignored if <code>interactive=TRUE</code> .
<code>phi</code>	See argument <code>theta</code> . Passed to the <code>phi</code> argument in the <code>persp</code> function. Ignored if <code>interactive=TRUE</code> .
<code>col</code>	The color(s) of the surface facets. Transparent colours are ignored. This is recycled to the $(n_x-1)(n_y-1)$ facets. Passed to the <code>col</code> argument in the <code>persp</code> function. Ignored if <code>interactive=TRUE</code> .
<code>shade</code>	The shade at a surface facet is computed as $((1+d)/2)^{\text{shade}}$ , where <code>d</code> is the dot product of a unit vector normal to the facet and a unit vector in the direction of a light source. Values of <code>shade</code> close to one yield shading similar to a point light source model and values close to zero produce no shading. Values in the range 0.5 to 0.75 provide an approximation to daylight illumination. Passed to the <code>shade</code> argument in the <code>persp</code> function. Ignored if <code>interactive=TRUE</code> .
<code>...</code>	Further arguments passed to <code>curve_cont</code> .

## Details

The survival curve or CIF dependent on a continuous variable can be viewed as a 3D surface. All other plots in this package try to visualize this surface by reducing it to two dimensions using color scales or summary statistics. This function on the other hand directly plots the 3D surface as such.

Although 3D plots are frowned upon by many scientists, it might be a good visualisation choice from time to time. Using the `interactive=TRUE` option makes the plot interactive, which lessens some of the valid criticism of 3D graphics. Similar looking plots for the same purpose but using a likelihood-ratio approach have been proposed in Smith et al. (2019).

## Value

Returns a plotly object if `interactive=TRUE` is used and a matrix when `interactive=FALSE` is used (still drawing the plot).

## Author(s)

Robin Denz

## References

Smith, A. M.; Christodouleas, J. P. & Hwang, W.-T. Understanding the Predictive Value of Continuous Markers for Censored Survival Data Using a Likelihood Ratio Approach BMC Medical Research Methodology, 2019, 19

## Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(splines)
library(ggplot2)
library(plotly)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(41)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on survival for ages 50 to 80
plot_surv_3Dsurface(time="futime",
                    status="status",
                    variable="age",
                    data=nafld1,
                    model=model,
                    horizon=seq(50, 80, 0.5),
                    interactive=FALSE)
```



```

# plot effect of age on survival using an interactive plot for ages 50 to 80
plot_surv_3Dsurface(time="fuptime",
                    status="status",
                    variable="age",
                    data=nafl1d1,
                    model=model,
                    horizon=seq(50, 80, 0.5),
                    interactive=TRUE)

## showing non-linear effects

# fit cox-model with bmi modeled using B-Splines,
# adjusting for age
model2 <- coxph(Surv(fuptime, status) ~ age + bs(bmi, df=3),
               data=nafl1d1, x=TRUE)

# plot effect of bmi on survival using normal plot
plot_surv_3Dsurface(time="fuptime",
                    status="status",
                    variable="bmi",
                    data=nafl1d1,
                    model=model2,
                    interactive=FALSE)

# plot effect of bmi on survival using interactive plot
plot_surv_3Dsurface(time="fuptime",
                    status="status",
                    variable="bmi",
                    data=nafl1d1,
                    model=model2,
                    interactive=TRUE)

```

---

plot_surv_animated	<i>Create an Animated Plot of Survival Curves of CIFs as a Function of a Continuous Variable</i>
--------------------	--

---

## Description

Using a previously fit time-to-event model, this function plots a single covariate-specific survival curve or CIF. Using either the **plotly** or the **gganimate** package, an animated plot is created that is either simply showing how the curve evolves over values of the continuous covariate, or includes a slide to let the user interact with it.

## Usage

```

plot_surv_animated(time, status, variable, group=NULL,
                  data, model, cif=FALSE,
                  na.action=options()$na.action,
                  horizon=NULL, fixed_t=NULL, max_t=Inf,

```

```

slider=TRUE, size=1, color="black",
linetype="solid", alpha=1,
xlab="Time", ylab="Survival Probability",
title=NULL, subtitle=NULL,
gg_theme=ggplot2::theme_bw(),
facet_args=list(), ...)

```

## Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to NULL (default) to use no grouping variable.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as a coxph model). Needs to include variable as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.
cif	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
horizon	A numeric vector containing a range of values of variable for which the survival curves should be calculated or NULL (default). If NULL, the horizon is constructed as a sequence from the lowest to the highest value observed in variable with 40 equally spaced steps.
fixed_t	A numeric vector containing points in time at which the survival probabilities should be calculated or NULL (default). If NULL, the survival probability is estimated at every point in time at which an event occurred.
max_t	A number indicating the latest survival time which is to be plotted.
slider	Whether to include a slider that controls the value of the continuous covariate or not. If not, the plot simply cycles through the values in horizon at a constant speed.

size	A single number specifying the size of the drawn curve.
color	A single character string specifying the color of the curve.
linetype	A single character string specifying the linetype of the curve.
alpha	The transparency level of the plot.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.
title	A character string used as the title of the plot. When <code>slider=FALSE</code> is used, the title is also used in a dynamic way to show which value the continuous variable currently has. The user-defined title is still shown, but the covariate level is also appended to it in that case.
subtitle	A character string used as the subtitle of the plot.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.
facet_args	A named list of arguments that are passed to the <code>facet_wrap</code> function call when creating a plot separated by groups. Ignored if <code>group=NULL</code> . Any argument except the <code>facets</code> argument of the <code>facet_wrap</code> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .
...	Further arguments passed to <code>curve_cont</code> .

### Details

When the plot doesn't have to be on paper, it is also possible to use animation to represent how the survival curve or CIF changes when different values of a continuous covariate are used. There are two options included in this function. The first one is creating a plot with a slider that can be interactively manipulated by the user (argument `slider=TRUE`). The second option creates a .gif file in which the continuous variable changes constantly over time (argument `slider=FALSE`).

This type of plot is only useful when visualizing the effect on a computer. It is also important to keep in mind that these plots might look slightly different to users with different hardware and or software.

An additional interactive plot can be created using the `plot_surv_3Dsurface` function with `interactive=TRUE`.

### Value

Returns a plotly object if `slider=TRUE` and a gganim object if `slider=FALSE`.

### Author(s)

Robin Denz

### Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(splines)
library(ggplot2)
```

```
library(gganimate)
library(transformr)
library(plotly)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

if (interactive()) {
# plot effect of age on survival for ages 50 to 80
plot_surv_animated(time="futime",
                   status="status",
                   variable="age",
                   data=nafld1,
                   model=model,
                   horizon=seq(50, 80, 1),
                   slider=TRUE)

# plot effect of age on survival using an interactive plot for ages 50 to 80
plot_surv_animated(time="futime",
                   status="status",
                   variable="age",
                   data=nafld1,
                   model=model,
                   horizon=seq(50, 80, 1),
                   slider=FALSE)
}

## showing non-linear effects

# fit cox-model with bmi modeled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(futime, status) ~ age + male + bs(bmi, df=3),
               data=nafld1, x=TRUE)

if (interactive()) {
# plot effect of bmi on survival using slider
plot_surv_animated(time="futime",
                   status="status",
                   variable="bmi",
                   data=nafld1,
                   model=model2,
                   slider=TRUE)

# plot effect of bmi on survival using a .gif
plot_surv_animated(time="futime",
                   status="status",
```

```

        variable="bmi",
        data=nafl1d1,
        model=model2,
        slider=FALSE)
}

```

---

plot_surv_area	<i>Plot a Survival Area Plot for the Effect of a Continuous Variable on a Time-To-Event Outcome</i>
----------------	---

---

### Description

Using a previously fit time-to-event model, this function plots a survival curve or CIF area plot. Instead of plotting value-specific curves, the probability of interest is represented as an area as a function of time, where the color changes according to the continuous variable.

### Usage

```

plot_surv_area(time, status, variable, group=NULL,
               data, model, cif=FALSE,
               na.action=options()$na.action,
               horizon=NULL, fixed_t=NULL, max_t=Inf,
               start_color="blue", end_color="red", alpha=1,
               discrete=FALSE, bins=ifelse(discrete, 10, 40),
               sep_lines=FALSE, sep_color="black",
               sep_size=0.1, sep_linetype="solid",
               sep_alpha=alpha, xlab="Time",
               ylab="Survival Probability", title=NULL,
               subtitle=NULL, legend.title=variable,
               legend.position="right",
               gg_theme=ggplot2::theme_bw(),
               facet_args=list(),
               label_digits=NULL,
               ...)

```

### Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.

group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to NULL (default) to use no grouping variable.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as an coxph model). Needs to include <code>variable</code> as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.
cif	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
horizon	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or NULL (default). If NULL, the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 40 equally spaced steps.
fixed_t	A numeric vector containing points in time at which the survival probabilities should be calculated or NULL (default). If NULL, the survival probability is estimated at every point in time at which an event occurred.
max_t	A number indicating the latest survival time which is to be plotted.
start_color	The color used for the lowest value in horizon. This and the <code>end_color</code> argument can be used to specify custom continuous color scales used in the plot. For example, if a black and white plot is desired, the user can set <code>start_color="white"</code> and <code>end_color="black"</code> . See <code>?scale_color_gradient</code> for more information.
end_color	The color used for the highest value in horizon. See argument <code>start_color</code> .
alpha	The transparency level of the main plot.
discrete	Whether to plot the area as a single continuously shaded area (default) or as multiple discrete blocks. The blocks correspond to ranges (such as 5-10). This only really changes the style of the legend and decreases the default value of bins to make the discrete blocks visible.
bins	The number of bins the survival area should be divided into. In a nutshell, this function simply estimates a lot of covariate specific curves and colors the area between them on a graded scale. If many bins are used, this gives the appearance of a single shaded area, changing continuously. By using only a few bins however, the specific covariate values might be easier to read off the plot. See also the <code>discrete</code> argument.
sep_lines	Whether to draw lines between the individual area segments or not. When <code>discrete=TRUE</code> is used, this might be a good option to help visually separate the segments.

sep_color	The color of the separator lines. Ignored if sep_lines=FALSE.
sep_size	The size of the separator lines. Ignored if sep_lines=FALSE.
sep_linetype	The linetype of the separator lines. Ignored if sep_lines=FALSE.
sep_alpha	The transparency level of the separator lines. Ignored if sep_lines=FALSE.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.
title	A character string used as the title of the plot.
subtitle	A character string used as the subtitle of the plot.
legend.title	A character string used as the legend title of the plot.
legend.position	Where to put the legend. See ?theme for more details.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.
facet_args	A named list of arguments that are passed to the <a href="#">facet_wrap</a> function call when creating a plot separated by groups. Ignored if group=NULL. Any argument except the facets argument of the <a href="#">facet_wrap</a> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .
label_digits	A single number specifying to how many digits the labels of the plot should be rounded or NULL (default). If NULL, no rounding is performed and the exact values are presented. This argument is only used when discrete=TRUE is used and ignored otherwise.
...	Further arguments passed to <a href="#">curve_cont</a> .

## Details

This function is very similar to a contour plot (see [plot\\_surv\\_contour](#)), but with the probability of interest on the y-axis and the color representing the values of the continuous variable of interest. The main advantage of this type of plot is, that it has the same structure as a usual Kaplan-Meier plot. The only difference is that instead of single curves stratified by some variable, the curves are plotted as an area instead.

This is achieved by estimating value-specific curves over the whole range of the continuous covariate using direct standardization and a previously fitted time-to-event model (see [curve\\_cont](#)). The area between those curves is then simply filled in (using the `geom_stepribbon` function from the **pammtools** package). By using a big number of individual curves, the appearance of a continuously shaded area is created. This is done by default when using the argument `discrete=FALSE`. By using `discrete=TRUE`, only a small number of value-specific curves are estimated, resulting in only a few discrete bins. When using only a few bins, the output is therefore even closer to a contour plot.

The major downside of this plotting method is that it only works if the effect of the continuous covariate was modeled as a monotone effect, meaning that it either only increases or only decreases with higher values. Curved relationships between the variable and the survival time cannot be represented using this function, because the areas would be plotted on top of each other in that case. However, the [plot\\_surv\\_contour](#) and [plot\\_surv\\_heatmap](#) functions may be used instead in this case.

**Value**

Returns a ggplot2 object.

**Author(s)**

Robin Denz

**References**

Robin Denz, Nina Timmesfeld (2022). Visualizing the Causal Effect of a Continuous Variable on a Time-To-Event Outcome. arXiv:2208.04644v1

**Examples**

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(pammtools)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 100), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on survival using defaults
plot_surv_area(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model)

# plot it only for 60 to 80 year old people
plot_surv_area(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               horizon=seq(60, 80, 0.5))

# plot it only for 60 to 60 year old people, using discrete bins
# and a black and white color scale
plot_surv_area(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
```



```
horizon=seq(60, 80, 5),
discrete=TRUE,
start_color="grey",
end_color="black")
```

---

plot\_surv\_at\_t                    *Plot the Survival Probability or CIF at a Fixed Point in Time as a Function of a Continuous Variable*

---

### Description

Using a previously fit time-to-event model, this function plots the survival probability or CIF at one or multiple user defined points at time as a function of a continuous variable.

### Usage

```
plot_surv_at_t(time, status, variable, group=NULL,
               data, model, cif=FALSE,
               na.action=options()$na.action, t, horizon=NULL,
               size=1, linetype="solid", alpha=1,
               xlab=variable, ylab="Survival Probability at t",
               title=NULL, subtitle=NULL,
               legend.title="t", legend.position="right",
               gg_theme=ggplot2::theme_bw(),
               facet_args=list(), ...)
```

### Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to NULL (default) to use no grouping variable.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as an coxph model). Needs to include variable as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.

<code>cif</code>	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
<code>na.action</code>	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
<code>t</code>	The point in time at which the survival probability should be calculated. For example, by setting this parameter to 10, this function will display the survival probability at $t = 10$ over values of <code>variable</code> . If multiple values are supplied, one curve is drawn for each of them.
<code>horizon</code>	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 100 equally spaced steps.
<code>size</code>	A single number specifying how thick the lines should be drawn.
<code>linetype</code>	The linetype of the drawn lines. See documentation of <b>ggplot2</b> for more details on allowed values.
<code>alpha</code>	The transparency level of the lines.
<code>xlab</code>	A character string used as the x-axis label of the plot.
<code>ylab</code>	A character string used as the y-axis label of the plot.
<code>title</code>	A character string used as the title of the plot.
<code>subtitle</code>	A character string used as the subtitle of the plot.
<code>legend.title</code>	A character string used as the legend title of the plot.
<code>legend.position</code>	Where to put the legend. See <code>?theme</code> for more details.
<code>gg_theme</code>	A <b>ggplot2</b> theme which is applied to the plot.
<code>facet_args</code>	A named list of arguments that are passed to the <code>facet_wrap</code> function call when creating a plot separated by groups. Ignored if <code>group=NULL</code> . Any argument except the <code>facets</code> argument of the <code>facet_wrap</code> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .
<code>...</code>	Further arguments passed to <code>curve_cont</code> .

## Details

By picking a single value of time, we can effectively reduce the dimensions of the plot by one, resulting in a simple curve plot of the survival probability or CIF at the specific point in time as it changes over values of the continuous covariate. By supplying a vector of times to the `t` argument, multiple such curves can be produced in the same plot.

This can be a very effective plotting strategy if there are specific points in time of interest already (even better if those were pre-defined). It is however a poor strategy if a representation of the effect

on the survival over time is desired. An empirical example of this strategy using the 5 year risk is given in Shen et al. (2017).

As all plot functions in this package, this plot relies on the [curve\\_cont](#) function to calculate the needed probability estimates.

### Value

Returns a ggplot2 object.

### Author(s)

Robin Denz

### References

Shen, Y.-M.; Le, L. D.; Wilson, R. & Mansmann, U. Graphical Presentation of Patient-Treatment Interaction Elucidated by Continuous Biomarkers: Current Practice and Scope for Improvement *Methods of Information in Medicine*, 2017, 56, 13-27

### Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on survival at t=2000
plot_surv_at_t(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               t=2000)

# plot it for arbitrary multiple values of t
plot_surv_at_t(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               t=c(1000, 2000, 3200, 5643))
```

---

plot_surv_contour	<i>Create a Contour Plot of the Effect of a Continuous Covariate on the Survival Probability</i>
-------------------	--

---

## Description

Using a previously fit time-to-event model, this function creates a contour plot with the continuous covariate on the y-axis and the time-to-event on the x-axis. The color is made in accordance with the corresponding survival probability at that point, binned into separate categories. This is very similar to the [plot\\_surv\\_heatmap](#) plot, but using a categorical representation.

## Usage

```
plot_surv_contour(time, status, variable, group=NULL,
                  data, model, cif=FALSE,
                  na.action=options()$na.action,
                  horizon=NULL, fixed_t=NULL, max_t=Inf,
                  size=0.1, linetype="solid", alpha=1,
                  bins=NULL, binwidth=NULL, breaks=NULL,
                  custom_colors=NULL,
                  xlab="Time", ylab=variable,
                  title=NULL, subtitle=NULL,
                  legend.title="S(t)", legend.position="right",
                  gg_theme=ggplot2::theme_bw(),
                  facet_args=list(),
                  panel_border=FALSE, axis_dist=0, ...)
```

## Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to NULL (default) to use no grouping variable.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as an coxph model). Needs to include variable as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.

<code>cif</code>	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
<code>na.action</code>	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
<code>horizon</code>	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 40 equally spaced steps.
<code>fixed_t</code>	A numeric vector containing points in time at which the survival probabilities should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the survival probability is estimated at 100 equally spaced steps from 0 to the maximum observed event time.
<code>max_t</code>	A number indicating the latest survival time which is to be plotted.
<code>size</code>	The size of the individual lines plotted by the <code>geom_contour_filled</code> function. Usually this option can be kept at 0.1 (default).
<code>linetype</code>	The linetype of the contour lines.
<code>alpha</code>	The transparency level of the plot.
<code>bins</code>	A single number specifying how many categories the survival probability or CIF should be divided in, or <code>NULL</code> (default). If <code>NULL</code> , the defaults in <code>geom_contour_filled</code> are used.
<code>binwidth</code>	A single number specifying the width of the bins of the survival probability or CIF categories, or <code>NULL</code> (default). If <code>NULL</code> , the defaults in <code>geom_contour_filled</code> are used.
<code>breaks</code>	A vector of specific breaks to create the survival probability or CIF bins, or <code>NULL</code> (default). If <code>NULL</code> , the defaults in <code>geom_contour_filled</code> are used.
<code>custom_colors</code>	A character vector of custom colors that should be used for each contour area.
<code>xlab</code>	A character string used as the x-axis label of the plot.
<code>ylab</code>	A character string used as the y-axis label of the plot.
<code>title</code>	A character string used as the title of the plot.
<code>subtitle</code>	A character string used as the subtitle of the plot.
<code>legend.title</code>	A character string used as the legend title of the plot.
<code>legend.position</code>	Where to put the legend. See <code>?theme</code> for more details.
<code>gg_theme</code>	A <b>ggplot2</b> theme which is applied to the plot.
<code>facet_args</code>	A named list of arguments that are passed to the <code>facet_wrap</code> function call when creating a plot separated by groups. Ignored if <code>group=NULL</code> . Any argument except the <code>facets</code> argument of the <code>facet_wrap</code> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .

panel_border	Whether to draw a border around the heatmap or not. Is set to FALSE by default to mimic standard heatmaps.
axis_dist	The distance of the axis ticks to the colored heatmap. Is set to 0 by default to mimic standard heatmaps.
...	Further arguments passed to <a href="#">curve_cont</a> .

### Details

Contour plots are a great way to reduce three dimensions to a two-dimensional plot and therefore lend themselves easily to plot the effect of continuous variable on the survival probability or CIF over time. Heatmaps (as available in the [plot\\_surv\\_heatmap](#)) function are very similar, but are often less clear than contour plots. The major downside of these plots is that they do not have the survival probability (or CIF) on the y-axis, which is of course the standard in standard Kaplan-Meier plots. A very similar plot but with the probability of interest on the y-axis can be created using the [plot\\_surv\\_area](#) function.

This type of plot has been described in Jackson & Cox (2021). However, the authors of that article propose a different method to estimate the needed data for the plot.

Internally, this function uses the [geom\\_contour\\_filled](#) function of the **ggplot2** package to create the plot, after estimating the required probabilities using the [curve\\_cont](#) function.

### Value

Returns a ggplot2 object.

### Author(s)

Robin Denz

### References

Robin Denz, Nina Timmesfeld (2022). Visualizing the Causal Effect of a Continuous Variable on a Time-To-Event Outcome. arXiv:2208.04644v1

Jackson, R. J. & Cox, T. F. Kernel Hazard Estimation for Visualisation of the Effect of a Continuous Covariate on Time-To-Event Endpoints Pharmaceutical Statistics, 2021

### Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(splines)

# using data from the survival package
data(naf1d, package="survival")

# take a random sample to keep example fast
set.seed(42)
naf1d1 <- naf1d1[sample(nrow(naf1d1), 150), ]
```

```

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafl1, x=TRUE)

# plot effect of age on survival using defaults
plot_surv_contour(time="futime",
                  status="status",
                  variable="age",
                  data=nafl1,
                  model=model)

# plot it only for 60 to 80 year old people
plot_surv_contour(time="futime",
                  status="status",
                  variable="age",
                  data=nafl1,
                  model=model,
                  horizon=seq(60, 80, 0.5))

## showing non-linear effects

# fit cox-model with bmi modelled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(futime, status) ~ age + male + bs(bmi, df=3),
               data=nafl1, x=TRUE)

# plot effect of bmi on survival using defaults
plot_surv_contour(time="futime",
                  status="status",
                  variable="bmi",
                  data=nafl1,
                  model=model2)

```

---

plot_surv_heatmap	<i>Plot a Heatmap of the Effect of a Continuous Covariate on a Time-To-Event Outcome</i>
-------------------	--

---

## Description

Using a previously fit time-to-event model, this function plots a heatmap with the continuous covariate on the y-axis and the time-to-event on the x-axis. The color is made in accordance with the corresponding survival probability or CIF at that point.

## Usage

```

plot_surv_heatmap(time, status, variable, group=NULL,
                  data, model, cif=FALSE,
                  na.action=options()$na.action,
                  horizon=NULL, fixed_t=NULL, max_t=Inf,
                  start_color=NULL, end_color=NULL,

```

```

alpha=1, xlab="Time", ylab=variable,
title=NULL, subtitle=NULL,
legend.title="S(t)", legend.position="right",
gg_theme=ggplot2::theme_bw(),
facet_args=list(), panel_border=FALSE,
axis_dist=0, interpolate=TRUE,
contour_lines=FALSE, contour_color="white",
contour_size=0.3, contour_linetype="dashed",
...)
```

## Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to NULL (default) to use no grouping variable.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as an coxph model). Needs to include variable as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See ?predictRisk for more details.
cif	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
na.action	How missing values should be handled. Can be one of: na.fail, na.omit, na.pass, na.exclude or a user-defined custom function. Also accepts strings of the function names. See ?na.action for more details. By default it uses the na.action which is set in the global options by the respective user.
horizon	A numeric vector containing a range of values of variable for which the survival curves should be calculated or NULL (default). If NULL, the horizon is constructed as a sequence from the lowest to the highest value observed in variable with 40 equally spaced steps.
fixed_t	A numeric vector containing points in time at which the survival probabilities should be calculated or NULL (default). If NULL, the survival probability is estimated at 100 equally spaced steps from 0 to the maximum observed event time.
max_t	A number indicating the latest survival time which is to be plotted.



start_color	The color used for the lowest value in horizon. This and the end_color argument can be used to specify custom continuous color scales used in the plot. For example, if a black and white plot is desired, the user can set start_color="white" and end_color="black". See ?scale_color_gradient for more information.
end_color	The color used for the highest value in horizon. See argument start_color.
alpha	The transparency level of the plot.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.
title	A character string used as the title of the plot.
subtitle	A character string used as the subtitle of the plot.
legend.title	A character string used as the legend title of the plot.
legend.position	Where to put the legend. See ?theme for more details.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.
facet_args	A named list of arguments that are passed to the <a href="#">facet_wrap</a> function call when creating a plot separated by groups. Ignored if group=NULL. Any argument except the facets argument of the <a href="#">facet_wrap</a> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to list(scales="free_y").
panel_border	Whether to draw a border around the heatmap or not. Is set to FALSE by default to mimic standard heatmaps.
axis_dist	The distance of the axis ticks to the colored heatmap. Is set to 0 by default to mimic standard heatmaps.
interpolate	Whether to linearly interpolate the colors or not. Set to TRUE by default, which results in a smooth surface being plotted. Corresponds to the interpolate argument in the geom_raster function, which is used internally.
contour_lines	Whether to add some contour lines to the heatmap. To get a proper contour plot, use the <a href="#">plot_surv_contour</a> function instead.
contour_color	The color of the contour lines. Defaults to "white". Ignored if contour_lines=FALSE.
contour_size	The size of the contour lines. Ignored if contour_lines=FALSE.
contour_linetype	The linetype of the contour lines. Defaults to "dashed". Ignored if contour_lines=FALSE.
...	Further arguments passed to <a href="#">curve_cont</a> .

## Details

Heatmaps are a great tool to visualize a three dimensional surface in a two-dimensional plot. A continuous color scale is used to represent the probability of interest. Although this is fine theoretically, it is often hard to read specific information off these plots. Contour lines can be added to the plot in order to make this easier by using contour\_lines=TRUE in the function call.

In most cases, however, it is probably better to use a proper contour plot instead, which can be produced using the [plot\\_surv\\_contour](#) function. This is mostly a matter of taste, which is why both types of plots are included in this package.

The main advantage of the heatmap and the contour plots is that they can visualize the effect of a continuous covariate regardless of how it was modeled. Non-linear relationships can be visualized just as well as linear ones. The major downside is, that the structure of the plot is not the same as that of a standard Kaplan-Meier plot. An alternative that is closer to the standard plot can be created using the `plot_surv_area` function.

### Value

Returns a ggplot2 object.

### Author(s)

Robin Denz

### Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(splines)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on survival using defaults
plot_surv_heatmap(time="futime",
                  status="status",
                  variable="age",
                  data=nafld1,
                  model=model)

# plot it only for 60 to 80 year old people
plot_surv_heatmap(time="futime",
                  status="status",
                  variable="age",
                  data=nafld1,
                  model=model,
                  horizon=seq(60, 80, 0.5))

## showing non-linear effects

# fit cox-model with bmi modelled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(futime, status) ~ age + male + bs(bmi, df=3),
```

```

data=nafl1d1, x=TRUE)

# plot effect of bmi on survival using defaults
plot_surv_heatmap(time="fuptime",
                  status="status",
                  variable="bmi",
                  data=nafl1d1,
                  model=model2)

# plot effect of bmi on survival with contour lines
plot_surv_heatmap(time="fuptime",
                  status="status",
                  variable="bmi",
                  data=nafl1d1,
                  model=model2,
                  contour_lines=TRUE)

```

---

plot_surv_lines	<i>Plot Individual Survival Curves or CIFs for Specific Values of a Continuous Covariate</i>
-----------------	--

---

### Description

Using a previously fit time-to-event model, this function plots survival curves or CIFs that would have been observed if every individual in the dataset had been set to specific values of a continuous covariate.

### Usage

```

plot_surv_lines(time, status, variable, group=NULL,
               data, model, cif=FALSE,
               na.action=options()$na.action,
               horizon=NULL, fixed_t=NULL, max_t=Inf,
               discrete=TRUE, custom_colors=NULL,
               start_color="blue", end_color="red",
               size=1, linetype="solid", alpha=1,
               xlab="Time", ylab="Survival Probability",
               title=NULL, subtitle=NULL,
               legend.title=variable, legend.position="right",
               gg_theme=ggplot2::theme_bw(),
               facet_args=list(), ...)

```

### Arguments

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.

variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the <code>data.frame</code> that is supplied to the <code>data</code> argument.
group	An optional single character string specifying a factor variable in <code>data</code> . When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to <code>NULL</code> (default) to use no grouping variable.
data	A <code>data.frame</code> containing all required variables.
model	A model describing the time-to-event process (such as a <code>coxph</code> model). Needs to include <code>variable</code> as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.
cif	Whether to plot the cumulative incidence (CIF) instead of the survival probability. If multiple failure types are present, the survival probability cannot be estimated in an unbiased way. This function will always return CIF estimates in that case.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
horizon	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 12 equally spaced steps.
fixed_t	A numeric vector containing points in time at which the survival probabilities should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the survival probability is estimated at every point in time at which an event occurred.
max_t	A number indicating the latest survival time which is to be plotted.
discrete	Whether to use a continuous color scale or a discrete one (default). If <code>FALSE</code> , the default <b>ggplot2</b> colors are used.
custom_colors	An optional character vector of colors to use when <code>discrete=FALSE</code> . Ignored if <code>discrete=TRUE</code> , in which case the color gradient can be defined using the <code>start_color</code> and <code>end_color</code> arguments.
start_color	The color used for the lowest value in <code>horizon</code> . This and the <code>end_color</code> argument can be used to specify custom continuous color scales used in the plot. For example, if a black and white plot is desired, the user can set <code>start_color="white"</code> and <code>end_color="black"</code> . See <code>?scale_color_gradient</code> for more information.
end_color	The color used for the highest value in <code>horizon</code> . See argument <code>start_color</code> .
size	A single number specifying the size of the drawn curves.
linetype	A single character string specifying the linetype of the curves.
alpha	The transparency level of the plot.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.

title	A character string used as the title of the plot.
subtitle	A character string used as the subtitle of the plot.
legend.title	A character string used as the legend title of the plot.
legend.position	Where to put the legend. See ?theme for more details.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.
facet_args	A named list of arguments that are passed to the <code>facet_wrap</code> function call when creating a plot separated by groups. Ignored if <code>group=NULL</code> . Any argument except the <code>facets</code> argument of the <code>facet_wrap</code> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .
...	Further arguments passed to <code>curve_cont</code> .

### Details

A simple plot of multiple covariate-specific survival curves. Internally, it uses the `curve_cont` function to calculate the survival curves.

### Value

Returns a `ggplot2` object.

### Author(s)

Robin Denz

### Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(splines)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on survival using defaults
plot_surv_lines(time="futime",
                status="status",
                variable="age",
                data=nafld1,
```

```

        model=model)

# plot it only for some specific user-defined values
plot_surv_lines(time="fuptime",
               status="status",
               variable="age",
               data=nafl1d1,
               model=model,
               horizon=c(40, 52, 63, 81))

## showing non-linear effects

# fit cox-model with bmi modelled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(fuptime, status) ~ age + male + bs(bmi, df=3),
               data=nafl1d1, x=TRUE)

# plot effect of bmi on survival
plot_surv_lines(time="fuptime",
               status="status",
               variable="bmi",
               data=nafl1d1,
               model=model2,
               horizon=c(20, 30, 40))

```

---

plot\_surv\_quantiles     *Plot Survival Time Quantiles as a Function of a Continuous Variable*

---

## Description

Using a previously fit time-to-event model, this function plots one or multiple survival time quantiles (such as the median survival time) as a function of a continuous variable of interest.

## Usage

```

plot_surv_quantiles(time, status, variable, group=NULL,
                   data, model, na.action=options()$na.action,
                   p=0.5, horizon=NULL,
                   size=1, linetype="solid", alpha=1,
                   custom_colors=NULL, single_color=NULL,
                   xlab=variable, ylab="Survival Time Quantile",
                   title=NULL, subtitle=NULL,
                   legend.title=variable, legend.position="right",
                   gg_theme=ggplot2::theme_bw(),
                   facet_args=list(), ...)

```

## Arguments

**time**                    A single character string specifying the time-to-event variable. Needs to be a valid column name of a numeric variable in data.

status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a numeric or logical variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the <code>data.frame</code> that is supplied to the <code>data</code> argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to <code>NULL</code> (default) to use no grouping variable.
data	A <code>data.frame</code> containing all required variables.
p	A numeric vector containing the survival time quantiles of interest. For example, if the user is interested in plotting only the median survival time $p=0.5$ should be used. When multiple values are supplied, one curve is drawn for each quantile.
model	A model describing the time-to-event process (such as a <code>coxph</code> model). Needs to include <code>variable</code> as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
horizon	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 100 equally spaced steps.
size	A single number specifying how thick the lines should be drawn.
linetype	The linetype of the drawn lines. See documentation of <b>ggplot2</b> for more details on allowed values.
alpha	The transparency level of the lines.
custom_colors	An optional character vector specifying the colors that should be used when multiple quantiles were supplied to the <code>p</code> argument. To set the whole plot or a single curve to one color only, use the <code>single_color</code> argument instead.
single_color	A single character string specifying the color of all drawn lines.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.
title	A character string used as the title of the plot.
subtitle	A character string used as the subtitle of the plot.
legend.title	A character string used as the legend title of the plot.
legend.position	Where to put the legend. See <code>?theme</code> for more details.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.

facet_args	A named list of arguments that are passed to the <code>facet_wrap</code> function call when creating a plot separated by groups. Ignored if <code>group=NULL</code> . Any argument except the <code>facets</code> argument of the <code>facet_wrap</code> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .
...	Further arguments passed to <code>curve_cont</code> .

## Details

Survival Time Quantiles are a single value summarizing the entire survival curve. For example, the most prominently used survival time quantile is the median survival time, which can be interpreted as the time at which half of the people in the sample are expected to have experienced the event of interest. This plot shows one or more of these quantiles as a function of a continuous variable of interest.

To calculate the survival time quantiles, it first calls the `curve_cont` function to get estimates of the value-specific survival curves. Afterwards, it uses step function interpolation to read off the survival time quantile from the estimates.

Although this is a simple way to plot the effect of a continuous covariate on the survival, it can give a misleading visualization of the relationship in some situations. Plots that do not use summary statistics, such as the `plot_surv_contour` and `plot_surv_area` plots, may be preferable.

## Value

Returns a `ggplot2` object.

## Author(s)

Robin Denz

## Examples

```
library(consurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(splines)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on the median survival time
plot_surv_quantiles(time="futime",
                    status="status",
```



```

        variable="age",
        data=nafl1d1,
        model=model)

# plot multiple survival time quantiles
plot_surv_quantiles(time="fuptime",
                    status="status",
                    variable="age",
                    data=nafl1d1,
                    model=model,
                    p=c(0.1, 0.25, 0.5, 0.75, 0.9))

## showing non-linear effects

# fit cox-model with bmi modelled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(fuptime, status) ~ age + male + bs(bmi, df=3),
               data=nafl1d1, x=TRUE)

# plot effect of bmi on survival
plot_surv_quantiles(time="fuptime",
                    status="status",
                    variable="bmi",
                    data=nafl1d1,
                    model=model2,
                    p=c(0.1, 0.25, 0.5, 0.75, 0.9))

```

---

plot_surv_rmst	<i>Plot the Effect of a Continuous Variable on the Restricted Mean Survival Time</i>
----------------	--

---

## Description

Using a previously fit time-to-event model, this function plots the restricted mean survival time (RMST) as a function of a continuous variable.

## Usage

```

plot_surv_rmst(time, status, variable, group=NULL,
               data, model, na.action=options()$na.action,
               tau, horizon=NULL, custom_colors=NULL,
               size=1, linetype="solid", alpha=1, color="black",
               xlab=variable, ylab="Restricted Mean Survival Time",
               title=NULL, subtitle=NULL,
               legend.title=variable, legend.position="right",
               gg_theme=ggplot2::theme_bw(),
               facet_args=list(), ...)

```

**Arguments**

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a variable in data.
variable	A single character string specifying the continuous variable of interest, for which the survival curves should be estimated. This variable has to be contained in the <code>data.frame</code> that is supplied to the <code>data</code> argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to <code>NULL</code> (default) to use no grouping variable.
data	A <code>data.frame</code> containing all required variables.
model	A model describing the time-to-event process (such as an <code>coxph</code> model). Needs to include <code>variable</code> as an independent variable. It also has to have an associated <code>predictRisk</code> method. See <code>?predictRisk</code> for more details.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
tau	The point in time to which the RMST should be calculated. Can be a vector of numbers. If multiple values are supplied, one curve is drawn for each of them.
horizon	A numeric vector containing a range of values of <code>variable</code> for which the survival curves should be calculated or <code>NULL</code> (default). If <code>NULL</code> , the horizon is constructed as a sequence from the lowest to the highest value observed in <code>variable</code> with 100 equally spaced steps.
custom_colors	An optional character vector of colors to use when there are multiple values in <code>tau</code> . Ignored if <code>length(tau)==1</code> , in which case the <code>color</code> argument can be used to specify the color of the single line.
size	A single number specifying how thick the lines should be drawn.
linetype	The linetype of the drawn lines. See documentation of <b>ggplot2</b> for more details on allowed values.
alpha	The transparency level of the lines.
color	The color of the curve if <code>tau</code> is a single value. If a numeric vector was supplied to <code>tau</code> , use the <code>custom_colors</code> argument to specify them instead.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.
title	A character string used as the title of the plot.
subtitle	A character string used as the subtitle of the plot.
legend.title	A character string used as the legend title of the plot.

legend.position	Where to put the legend. See <code>?theme</code> for more details.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.
facet_args	A named list of arguments that are passed to the <code>facet_wrap</code> function call when creating a plot separated by groups. Ignored if <code>group=NULL</code> . Any argument except the <code>facets</code> argument of the <code>facet_wrap</code> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to <code>list(scales="free_y")</code> .
...	Further arguments passed to <code>curve_cont</code> .

### Details

Similar to the `plot_surv_at_t` and `plot_surv_quantiles` plots, this function produces a plot of a survival curve summary statistic as a function of a continuous variable. The summary statistic in question is the restricted mean survival time (RMST), which is the area under a survival curve up to a specific point in time  $\tau$ . It can be interpreted as the mean survival time of the population up to  $\tau$ .

First, a range of value-specific survival curves are estimated. The RMST is then calculated for each one using step function interpolation. This only works for survival curves, not for CIFs. If the area under the CIF should be used instead, the `plot_surv_rmtl` function can be used.

An advantage of this method over the `plot_surv_at_t` and `plot_surv_quantiles` function is, that it does take the whole survival curve into account, kind of. Although it is the area under it, it is only calculated up to  $\tau$  which makes the output dependent on the choice of  $\tau$ . This, again, can result in deceiving plots in some cases. Plots visualizing the entire survival curves such as `plot_surv_contour` and `plot_surv_area` might be preferable.

### Value

Returns a ggplot2 object.

### Author(s)

Robin Denz

### References

Eng, K. H.; Schiller, E. & Morrell, K. On Representing the Prognostic Value of Continuous Gene Expression Biomarkers with the Restricted Mean Survival Curve. In: *Oncotarget*, 2015, 6, 36308-36318

### Examples

```
library(contsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(splines)

# using data from the survival package
```

```

data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld1[sample(nrow(nafld1), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on the RMST for ages 50 to 80
plot_surv_rmst(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               horizon=seq(50, 80, 1),
               tau=2500)

# plot RMST for multiple tau values for ages 50 to 80
plot_surv_rmst(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               horizon=seq(50, 80, 1),
               tau=c(2000, 3000, 5000))

## showing non-linear effects

# fit cox-model with bmi modeled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(futime, status) ~ age + male + bs(bmi, df=3),
               data=nafld1, x=TRUE)

# plot effect of bmi on survival
plot_surv_rmst(time="futime",
               status="status",
               variable="bmi",
               data=nafld1,
               model=model2,
               tau=c(2000, 3000, 5000))

```

---

plot\_surv\_rmtl

---

*Plot the Effect of a Continuous Variable on the Restricted Mean Time Lost*


---

### Description

Using a previously fit time-to-event model, this function plots the restricted mean time lost (RMTL) as a function of a continuous variable. Unlike the `plot_surv_rmst` function, this function can deal with multiple event types in the status variable.

**Usage**

```
plot_surv_rmtl(time, status, variable, group=NULL,
               data, model, na.action=options()$na.action,
               tau, horizon=NULL, custom_colors=NULL,
               size=1, linetype="solid", alpha=1, color="black",
               xlab=variable, ylab="Restricted Mean Time Lost",
               title=NULL, subtitle=NULL,
               legend.title=variable, legend.position="right",
               gg_theme=ggplot2::theme_bw(),
               facet_args=list(), ...)
```

**Arguments**

time	A single character string specifying the time-to-event variable. Needs to be a valid column name of a variable in data.
status	A single character string specifying the status variable, indicating if a person has experienced an event or not. Needs to be a valid column name of a variable in data.
variable	A single character string specifying the continuous variable of interest, for which the CIFs should be estimated. This variable has to be contained in the data.frame that is supplied to the data argument.
group	An optional single character string specifying a factor variable in data. When used, the plot is created conditional on this factor variable, meaning that a faceted plot is produced with one facet for each level of the factor variable. See <a href="#">curve_cont</a> for a detailed description of the estimation strategy. Set to NULL (default) to use no grouping variable.
data	A data.frame containing all required variables.
model	A model describing the time-to-event process (such as a coxph model). Needs to include variable as an independent variable. It also has to have an associated <a href="#">predictRisk</a> method. See <code>?predictRisk</code> for more details.
na.action	How missing values should be handled. Can be one of: <code>na.fail</code> , <code>na.omit</code> , <code>na.pass</code> , <code>na.exclude</code> or a user-defined custom function. Also accepts strings of the function names. See <code>?na.action</code> for more details. By default it uses the <code>na.action</code> which is set in the global options by the respective user.
tau	The point in time to which the RMTL should be calculated. Can be a vector of numbers. If multiple values are supplied, one curve is drawn for each of them.
horizon	A numeric vector containing a range of values of variable for which the CIFs should be calculated or NULL (default). If NULL, the horizon is constructed as a sequence from the lowest to the highest value observed in variable with 100 equally spaced steps.
custom_colors	An optional character vector of colors to use when there are multiple values in tau. Ignored if <code>length(tau)==1</code> , in which case the color argument can be used to specify the color of the single line.
size	A single number specifying how thick the lines should be drawn.

linetype	The linetype of the drawn lines. See documentation of <b>ggplot2</b> for more details on allowed values.
alpha	The transparency level of the lines.
color	The color of the curve if tau is a single value. If a numeric vector was supplied to tau, use the custom_colors argument to specify them instead.
xlab	A character string used as the x-axis label of the plot.
ylab	A character string used as the y-axis label of the plot.
title	A character string used as the title of the plot.
subtitle	A character string used as the subtitle of the plot.
legend.title	A character string used as the legend title of the plot.
legend.position	Where to put the legend. See ?theme for more details.
gg_theme	A <b>ggplot2</b> theme which is applied to the plot.
facet_args	A named list of arguments that are passed to the <a href="#">facet_wrap</a> function call when creating a plot separated by groups. Ignored if group=NULL. Any argument except the facets argument of the <a href="#">facet_wrap</a> function can be used. For example, if the user wants to allow free y-scales, this argument could be set to list(scales="free_y").
...	Further arguments passed to <a href="#">curve_cont</a> .

### Details

This function is essentially equal to the [plot\\_surv\\_rmtl](#) function. The only difference is that instead of using the restricted mean survival time (RMST), which is the area under the survival curve up to a specific point in time tau, this function uses the restricted mean time lost (RMTL), which is the area under the (cause-specific) CIF up to a specific point in time tau. It can be interpreted as the mean time it takes an individual to succumb to the event of interest before tau.

The reason this is split into two functions is, that the RMTL can be calculated when mutually exclusive competing events are present, while the RMST cannot. The basic pros and cons of the [plot\\_surv\\_rmst](#) function still apply here, however. For more information we suggest consulting the documentation page of the [plot\\_surv\\_rmst](#) function.

### Value

Returns a ggplot2 object.

### Author(s)

Robin Denz

### References

Eng, K. H.; Schiller, E. & Morrell, K. On Representing the Prognostic Value of Continuous Gene Expression Biomarkers with the Restricted Mean Survival Curve. In: *Oncotarget*, 2015, 6, 36308-36318

**Examples**

```
library(condsurvplot)
library(riskRegression)
library(survival)
library(ggplot2)
library(splines)

# using data from the survival package
data(nafld, package="survival")

# take a random sample to keep example fast
set.seed(42)
nafld1 <- nafld[sample(nrow(nafld), 150), ]

# fit cox-model with age
model <- coxph(Surv(futime, status) ~ age, data=nafld1, x=TRUE)

# plot effect of age on the RMST for ages 50 to 80
plot_surv_rmtl(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               horizon=seq(50, 80, 1),
               tau=2500)

# plot RMST for multiple tau values for ages 50 to 80
plot_surv_rmtl(time="futime",
               status="status",
               variable="age",
               data=nafld1,
               model=model,
               horizon=seq(50, 80, 1),
               tau=c(2000, 3000, 5000))

## showing non-linear effects

# fit cox-model with bmi modeled using B-Splines,
# adjusting for age and sex
model2 <- coxph(Surv(futime, status) ~ age + male + bs(bmi, df=3),
               data=nafld1, x=TRUE)

# plot effect of bmi on survival
plot_surv_rmtl(time="futime",
               status="status",
               variable="bmi",
               data=nafld1,
               model=model2,
               tau=c(2000, 3000, 5000))
```

# Index

contsurvplot-package, 2  
curve\_cont, 2, 3, 7, 10, 11, 14, 15, 17–20, 22,  
24, 25, 28, 29, 31, 32, 34, 35, 37, 38  
facet\_wrap, 11, 15, 18, 21, 25, 29, 32, 35, 38  
geom\_contour\_filled, 21, 22  
persp, 6  
plot\_surv\_3Dsurface, 6, 11  
plot\_surv\_animated, 9  
plot\_surv\_area, 2, 3, 13, 22, 26, 32, 35  
plot\_surv\_at\_t, 17, 35  
plot\_surv\_contour, 2, 3, 15, 20, 25, 32, 35  
plot\_surv\_heatmap, 2, 15, 20, 22, 23  
plot\_surv\_lines, 27  
plot\_surv\_quantiles, 30, 35  
plot\_surv\_rmst, 33, 36, 38  
plot\_surv\_rmtl, 35, 36, 38  
predictRisk, 4–6, 10, 14, 17, 20, 24, 28, 31,  
34, 37