

Package: `survinng` (via `r-universe`)

May 10, 2026

Title Gradient-Based Feature Attribution for Survival Neural Networks

Version 0.1.0

Description This package implements model-specific, gradient-based feature attribution methods for deep survival neural networks, including `DeepHit`, `CoxTime`, and `DeepSurv`. It accompanies the ICML 2025 paper ``Gradient-based Explanations for Deep Learning Survival Models".

License MIT + file LICENSE

Imports `checkmate`, `ggplot2`, `torch`, `dplyr`, `methods`, `utils`

Suggests `data.table`, `keras`, `knitr`, `rmarkdown`, `simsurv`, `ggmap`, `cowplot`, `survminer`, `torchvision`, `survival`, `survivalmodels`, `patchwork`, `here`, `survex`, `SurvMetrics`, `tensorflow`, `testthat` ($\geq 3.0.0$)

Config/testthat/edition 3

Encoding UTF-8

Roxygen `list(markdown = TRUE)`

RoxygenNote 7.2.3

Config/Needs/website `rmarkdown`

URL <https://bips-hb.github.io/survinng/>

VignetteBuilder `knitr`

Repository <https://bips-hb.r-universe.dev>

Date/Publication 2025-05-28 20:59:12 UTC

RemoteUrl <https://github.com/bips-hb/survinng>

RemoteRef HEAD

RemoteSha 37e2ae011468f92544c10c6fa1398af92d922be9

Contents

<code>as.data.frame.surv_result</code>	2
<code>explain</code>	2
<code>extract_model</code>	4

plot.surv_result	5
print.explainer_coxtime	6
print.extracted_survivalmodels_coxtime	7
print.surv_result	7
surv_grad	8
surv_gradSHAP	9
surv_intgrad	12
surv_smoothgrad	14

Index	16
--------------	-----------

as.data.frame.surv_result

Convert survival attribution results to a data.frame

Description

This function converts the survival attribution results into a data frame format. It can handle both stacked and non-stacked formats.

Usage

```
## S3 method for class 'surv_result'
as.data.frame(x, ..., stacked = FALSE)

as.data.table.surv_result(x, ..., stacked = FALSE)
```

Arguments

x	An object of class <code>surv_result</code> containing the survival attribution results.
...	Unused arguments.
stacked	Logical indicating whether to convert to a stacked data frame, i.e., the attributions are stacked on top of each other. Default is <code>FALSE</code> .

explain

Explain a model

Description

This function is a generic method that dispatches to the appropriate explain method based on the class of the model.

Usage

```

explain(
  model,
  data = NULL,
  model_type = NULL,
  baseline_hazard = NULL,
  labtrans = NULL,
  time_bins = NULL,
  preprocess_fun = NULL,
  postprocess_fun = NULL,
  predict_fun = NULL
)

## S3 method for class 'nn_module'
explain(
  model,
  data,
  model_type,
  baseline_hazard = NULL,
  labtrans = NULL,
  time_bins = NULL,
  preprocess_fun = NULL,
  postprocess_fun = NULL,
  predict_fun = NULL
)

## S3 method for class 'extracted_survivalmodels_coxtime'
explain(model, data, ...)

## S3 method for class 'extracted_survivalmodels_deephit'
explain(model, data, ...)

## S3 method for class 'extracted_survivalmodels_deepsurv'
explain(model, data, ...)

```

Arguments

<code>model</code>	A model object.
<code>data</code>	A data frame or matrix of data to explain the model.
<code>model_type</code>	A string specifying the type of the survival model. Possible values are "coxtime", "deephit", or "deepsurv".
<code>baseline_hazard</code>	A data frame containing the baseline hazard. It should have two columns: "time" and "hazard". This is only used for "coxtime" and "deepsurv" models.
<code>labtrans</code>	A list containing the transformation functions for the time variable. It should have two elements: "transform" and "transform_inv". This is highly experimental and not yet fully supported.

time_bins	A numeric vector specifying the time bins for the "deephit" model, e.g., <code>c(0, 1, 2, 3)</code> .
preprocess_fun	A function to preprocess the data before making predictions, e.g., adding a time variable for a coxtime model. This argument is highly experimental and the default values should work for most cases.
postprocess_fun	A function to postprocess the predictions after making them. This argument is highly experimental and the default values should work.
predict_fun	A function that can be used to make predictions from the model. If NULL, the predict method of the model will be used.
...	Unused arguments.

Value

An object of class explainer that contains the model, the data, and the prediction function.

extract_model	<i>Extract model information from a survivalmodels object</i>
---------------	---

Description

This function extracts model information from a neural network trained with the survivalmodels package.

Usage

```
extract_model(x, path = NULL, num_basehazard = 200L)

## S3 method for class 'coxtime'
extract_model(x, path = NULL, num_basehazard = 200L)

## S3 method for class 'deephit'
extract_model(x, path = NULL, ...)

## S3 method for class 'deepsurv'
extract_model(x, path = NULL, num_basehazard = 200L)
```

Arguments

x	A survivalmodels object, i.e., <code>survivalmodels::deephit</code> , <code>survivalmodels::coxtime</code> , or <code>survivalmodels::deepsurv</code> .
path	A string specifying the path to save the extracted model. If NULL, the model is not saved. Default is NULL.
num_basehazard	An integer specifying the number of points of the baseline hazard to compute. Default is 200L. This argument is only used for coxtime and deepsurv models.
...	Unused arguments.

plot.surv_result *Plot Methods for Survival Attribution Results*

Description

Visualize survival predictions, feature attributions, and contribution percentages and force plots for survival results. The latter two are specifically for GradSHAP(t) and IntGrad(t) methods.

Usage

```
## S3 method for class 'surv_result'
plot(x, ..., type = "attr")

plot_force(x, num_bars = 10)

plot_pred(x)

plot_attr(x, normalize = "none", add_comp = NULL)

plot_contr(x, aggregate = FALSE)
```

Arguments

x	An object of class <code>surv_result</code> containing survival attribution results.
...	(unused arguments)
type	Type of plot to generate when using the generic <code>plot()</code> method. Options: <ul style="list-style-type: none"> "pred": plot survival predictions over time "attr": plot feature attributions over time (default) "contr": plot feature contributions percentages over time "force": plot force plots for each instance
num_bars	Number of bars to show in the force plot. Default is 10.
normalize	Normalization method for <code>plot_attr()</code> . Options: <ul style="list-style-type: none"> "none" (default): no normalization "abs": normalize by the sum of absolute values "rel": normalize by the sum of values Note: Only recommended for visualization of GradSHAP(t) or IntGrad(t) results.
add_comp	Optional vector of comparison curves to add to the attribution plot (<code>plot_attr()</code> only). Options include: <ul style="list-style-type: none"> "pred": predicted survival curve "pred_ref": reference survival curve "pred_diff": difference between prediction and reference You can also specify "all" to include all three curves. Default is NULL.
aggregate	Logical; if TRUE, contributions are aggregated across all instances in <code>plot_contr()</code> . If FALSE (default), one panel per instance is shown.

Details

These functions provide a convenient way to visualize the results of survival attribution methods:

- `plot()` is a generic wrapper that dispatches to the appropriate plot type based on the `type` argument.
- `plot_pred()` visualizes survival predictions across time for the selected instances.
- `plot_attr()` displays time-resolved attributions over time per instance.

The following methods are only available for `GradSHAP(t)` and `IntGrad(t)`:

- `plot_contr()` visualizes the relative contribution of features over time, optionally aggregated across instances for global insights.
- `plot_force()` generates force plots showing the features' effect to the prediction over time.

Value

A `ggplot2` object.

```
print.explainer_coxtime
```

Custom print method for explainer objects

Description

This function prints a summary of the explainer object.

Usage

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

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

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

Arguments

`x` An object of class `'explainer_coxtime'`, `'explainer_deepsurv'`, or `'explainer_deephit'`.
`...` Additional arguments (not used).

```
print.extracted_survivalmodels_coxtime
```

Print method for extracted pycox survival model

Description

Print method for extracted pycox survival model

Usage

```
## S3 method for class 'extracted_survivalmodels_coxtime'  
print(x, ...)  
  
## S3 method for class 'extracted_survivalmodels_deepsurv'  
print(x, ...)  
  
## S3 method for class 'extracted_survivalmodels_deephit'  
print(x, ...)
```

Arguments

x	An object of class extracted_survivalmodels_coxtime, extracted_survivalmodels_deepsurv, or extracted_survivalmodels_deephit.
...	Additional arguments (not used).

```
print.surv_result
```

Print function for surv_result objects

Description

Print function for surv_result objects

Usage

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

Arguments

x	An object of class "surv_result"
...	Additional arguments (not used)

`surv_grad`*Calculate the Gradient of the Survival Function*

Description

This function calculates the gradient of the survival function with respect to the input features and time points for a given instance. In the paper, this is referred to as the "*Grad(t)*" method. It shows the sensitivity of the survival function to changes in the input features at a specific time point.

Usage

```
surv_grad(  
    exp,  
    target = "survival",  
    instance = 1,  
    times_input = FALSE,  
    batch_size = 50,  
    dtype = "float",  
    include_time = FALSE  
)  
  
## S3 method for class 'explainer_deepsurv'  
surv_grad(  
    exp,  
    target = "survival",  
    instance = 1,  
    times_input = FALSE,  
    batch_size = 1000,  
    dtype = "float",  
    ...  
)  
  
## S3 method for class 'explainer_coxtime'  
surv_grad(  
    exp,  
    target = "survival",  
    instance = 1,  
    times_input = FALSE,  
    batch_size = 1000,  
    dtype = "float",  
    include_time = FALSE  
)  
  
## S3 method for class 'explainer_deephit'  
surv_grad(  
    exp,  
    target = "survival",
```

```

instance = 1,
times_input = FALSE,
batch_size = 1000,
dtype = "float",
...
)

```

Arguments

exp	An object of class <code>explainer_deepsurv</code> , <code>explainer_coxtime</code> , or <code>explainer_deephit</code> .
target	A character string indicating the target output. For <code>DeepSurv</code> and <code>CoxTime</code> , it can be either "survival" (default), "cum_hazard", or "hazard". For <code>DeepHit</code> , it can be "survival" (default), "cif", or "pmf".
instance	An integer specifying the instance for which the gradient is calculated. It should be between 1 and the number of instances in the dataset.
times_input	A logical value indicating whether the gradient should be multiplied with input. In the paper, this variant is referred to as " $GxI(t)$ ".
batch_size	An integer specifying the batch size for processing. The default is 1000. This value describes the number of instances within one batch and not the final number of rows in the batch. For example, <code>CoxTime</code> replicates each instance for each time point.
dtype	A character string indicating the data type for the tensors. It can be either "float" (default) or "double".
include_time	A logical value indicating whether to include the time points in the output. This is only relevant for <code>CoxTime</code> and is ignored for <code>DeepSurv</code> and <code>DeepHit</code> .
...	Unused arguments.

See Also

Other Attribution Methods: [surv_gradSHAP\(\)](#), [surv_intgrad\(\)](#), [surv_smoothgrad\(\)](#)

surv_gradSHAP

Calculate the GradSHAP values of the Survival Function

Description

This function calculates the GradSHAP values of the survival function with respect to the input features and time points for a given instance. In the paper, this is referred to as the " $GradSHAP(t)$ " method. It is a fast and model-specific method for calculating the Shapley values for a deep survival model.

Usage

```
surv_gradSHAP(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = TRUE,  
  batch_size = 1000,  
  n = 50,  
  num_samples = 10,  
  data_ref = NULL,  
  dtype = "float",  
  replace = TRUE,  
  include_time = FALSE  
)  
  
## S3 method for class 'explainer_deepsurv'  
surv_gradSHAP(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = TRUE,  
  batch_size = 1000,  
  n = 50,  
  num_samples = 10,  
  data_ref = NULL,  
  dtype = "float",  
  replace = TRUE,  
  ...  
)  
  
## S3 method for class 'explainer_coxtime'  
surv_gradSHAP(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = TRUE,  
  batch_size = 1000,  
  n = 50,  
  num_samples = 10,  
  data_ref = NULL,  
  dtype = "float",  
  replace = TRUE,  
  include_time = FALSE  
)  
  
## S3 method for class 'explainer_deephit'  
surv_gradSHAP(  
  exp,
```

```

    target = "survival",
    instance = 1,
    times_input = TRUE,
    batch_size = 1000,
    n = 50,
    num_samples = 10,
    data_ref = NULL,
    dtype = "float",
    replace = TRUE,
    ...
)

```

Arguments

exp	An object of class <code>explainer_deepsurv</code> , <code>explainer_coxtime</code> , or <code>explainer_deephit</code> .
target	A character string indicating the target output. For <code>DeepSurv</code> and <code>CoxTime</code> , it can be either "survival" (default), "cum_hazard", or "hazard". For <code>DeepHit</code> , it can be "survival" (default), "cif", or "pmf".
instance	An integer specifying the instance for which the GradSHAP values are calculated. It should be between 1 and the number of instances in the dataset.
times_input	A logical value indicating whether the GradSHAP values should be multiplied with input.
batch_size	An integer specifying the batch size for processing. The default is 1000. This value describes the number of instances within one batch and not the final number of rows in the batch. For example, <code>CoxTime</code> replicates each instance for each time point.
n	An integer specifying the number of samples to be used for approximating the integral. The default is 50.
num_samples	An integer specifying the number of samples to be used for the baseline distribution. The default is 10.
data_ref	A reference dataset for sampling. If <code>NULL</code> , the reference dataset is taken from the input data of the model. This dataset should contain the same number of features as the input data.
dtype	A character string indicating the data type for the tensors. It can be either "float" (default) or "double".
replace	A logical value indicating whether to sample from the baseline distribution with replacement. The default is <code>TRUE</code> .
include_time	A logical value indicating whether to calculate GradSHAP also for each time point. This is only relevant for <code>CoxTime</code> and is ignored for <code>DeepSurv</code> and <code>DeepHit</code> . The default is <code>FALSE</code> .
...	Unused arguments. #' @return Returns an object of class <code>surv_result</code> .

See Also

Other Attribution Methods: [surv_grad\(\)](#), [surv_intgrad\(\)](#), [surv_smoothgrad\(\)](#)

`surv_intgrad`*Calculate the Integrated Gradients of the Survival Function*

Description

This function calculates the integrated gradients of the survival function with respect to the input features and time points for a given instance. In the paper, this is referred to as the *"IntGrad(t)"* method. It shows the attributions of the input features to the target function with respect to a reference input.

Usage

```
surv_intgrad(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = TRUE,  
  batch_size = 50,  
  n = 10,  
  x_ref = NULL,  
  dtype = "float",  
  include_time = FALSE  
)  
  
## S3 method for class 'explainer_deepsurv'  
surv_intgrad(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = TRUE,  
  batch_size = 1000,  
  n = 10,  
  x_ref = NULL,  
  dtype = "float",  
  ...  
)  
  
## S3 method for class 'explainer_coxtime'  
surv_intgrad(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = TRUE,  
  batch_size = 1000,  
  n = 10,  
  x_ref = NULL,  
  dtype = "float",
```

```

    include_time = FALSE
  )

  ## S3 method for class 'explainer_deephit'
  surv_intgrad(
    exp,
    target = "survival",
    instance = 1,
    times_input = TRUE,
    batch_size = 1000,
    n = 10,
    x_ref = NULL,
    dtype = "float",
    ...
  )

```

Arguments

<code>exp</code>	An object of class <code>explainer_deepsurv</code> , <code>explainer_coxtime</code> , or <code>explainer_deephit</code> .
<code>target</code>	A character string indicating the target output. For <code>DeepSurv</code> and <code>CoxTime</code> , it can be either "survival" (default), "cum_hazard", or "hazard". For <code>DeepHit</code> , it can be "survival" (default), "cif", or "pmf".
<code>instance</code>	An integer specifying the instance for which the integrated gradients are calculated. It should be between 1 and the number of instances in the dataset.
<code>times_input</code>	A logical value indicating whether the integrated gradients should be multiplied with input. Default is <code>TRUE</code> .
<code>batch_size</code>	An integer specifying the batch size for processing. The default is 1000. This value describes the number of instances within one batch and not the final number of rows in the batch. For example, <code>CoxTime</code> replicates each instance for each time point.
<code>n</code>	An integer specifying the number of approximation points for the integral calculation. Default is 10.
<code>x_ref</code>	A reference input for the integrated gradients. If <code>NULL</code> (default), the mean of the input data is used. It should have the same dimensions as the input data.
<code>dtype</code>	A character string indicating the data type for the tensors. It can be either "float" (default) or "double".
<code>include_time</code>	A logical value indicating whether to include attributions for the time points. This is only relevant for <code>CoxTime</code> and is ignored for <code>DeepSurv</code> and <code>DeepHit</code> .
<code>...</code>	Unused arguments.

See Also

Other Attribution Methods: [surv_gradSHAP\(\)](#), [surv_grad\(\)](#), [surv_smoothgrad\(\)](#)

`surv_smoothgrad`*Calculate the SmoothGrad values of the Survival Function*

Description

This function calculates the SmoothGrad values of the survival function with respect to the input features and time points for a given instance. In the paper, this is referred to as the " $SG(t)$ " method. It shows the smoothed sensitivity of the survival function to changes in the input features at a specific time point.

Usage

```
surv_smoothgrad(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = FALSE,  
  batch_size = 50,  
  n = 10,  
  noise_level = 0.1,  
  dtype = "float",  
  include_time = FALSE  
)  
  
## S3 method for class 'explainer_deepsurv'  
surv_smoothgrad(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = FALSE,  
  batch_size = 1000,  
  n = 10,  
  noise_level = 0.1,  
  dtype = "float",  
  ...  
)  
  
## S3 method for class 'explainer_coxtime'  
surv_smoothgrad(  
  exp,  
  target = "survival",  
  instance = 1,  
  times_input = FALSE,  
  batch_size = 1000,  
  n = 10,  
  noise_level = 0.1,  
  dtype = "float",
```

```

    include_time = FALSE
  )

  ## S3 method for class 'explainer_deephit'
  surv_smoothgrad(
    exp,
    target = "survival",
    instance = 1,
    times_input = FALSE,
    batch_size = 1000,
    n = 10,
    noise_level = 0.1,
    dtype = "float",
    ...
  )

```

Arguments

exp	An object of class <code>explainer_deepsurv</code> , <code>explainer_coxtime</code> , or <code>explainer_deephit</code> .
target	A character string indicating the target output. For <code>DeepSurv</code> and <code>CoxTime</code> , it can be either "survival" (default), "cum_hazard", or "hazard". For <code>DeepHit</code> , it can be "survival" (default), "cif", or "pmf".
instance	An integer specifying the instance for which the <code>SmoothGrad</code> is calculated. It should be between 1 and the number of instances in the dataset.
times_input	A logical value indicating whether the <code>SmoothGrad</code> should be multiplied with input. In the paper, this variant is referred to as " $SGxI(t)$ ".
batch_size	An integer specifying the batch size for processing. The default is 1000. This value describes the number of instances within one batch and not the final number of rows in the batch. For example, <code>CoxTime</code> replicates each instance for each time point.
n	An integer specifying the number of noise samples to be added to the input features. The default is 10.
noise_level	A numeric value specifying the level of Gaussian noise to be added to the input features. The default is 0.1.
dtype	A character string indicating the data type for the tensors. It can be either "float" (default) or "double".
include_time	A logical value indicating whether to also calculate the gradients with respect to the time. This is only relevant for <code>CoxTime</code> and is ignored for <code>DeepSurv</code> and <code>DeepHit</code> .
...	Unused arguments.

See Also

Other Attribution Methods: [surv_gradSHAP\(\)](#), [surv_grad\(\)](#), [surv_intgrad\(\)](#)

Index

* Attribution Methods

- surv_grad, 8
- surv_gradSHAP, 9
- surv_intgrad, 12
- surv_smoothgrad, 14

* Conversion Methods

- as.data.frame.surv_result, 2

* Plot Methods

- plot.surv_result, 5

as.data.frame.surv_result, 2

as.data.table.surv_result

(as.data.frame.surv_result), 2

explain, 2

extract_model, 4

plot.surv_result, 5

plot_attr(plot.surv_result), 5

plot_contr(plot.surv_result), 5

plot_force(plot.surv_result), 5

plot_pred(plot.surv_result), 5

print.explainer_coxtime, 6

print.explainer_deephit

(print.explainer_coxtime), 6

print.explainer_deepsurv

(print.explainer_coxtime), 6

print.extracted_survivalmodels_coxtime,

7

print.extracted_survivalmodels_deephit

(print.extracted_survivalmodels_coxtime),

7

print.extracted_survivalmodels_deepsurv

(print.extracted_survivalmodels_coxtime),

7

print.surv_result, 7

surv_grad, 8, 11, 13, 15

surv_gradSHAP, 9, 9, 13, 15

surv_intgrad, 9, 11, 12, 15

surv_smoothgrad, 9, 11, 13, 14