Package 'spect'

April 8, 2025

Version 1.0

Date 2025-04-06

Title Survival Prediction Ensemble Classification Tool

Author Stephen Abrams [aut, cre]

Maintainer Stephen Abrams <stephen.abrams@gmail.com>

Depends R (>= 4.0), futile.logger, dplyr

Imports doParallel, ggplot2, survminer, riskRegression, caret, caretEnsemble, survival, rlang

Description A tool for survival analysis using a discrete time approach with ensemble binary classification. 'spect' provides a simple interface consistent with commonly used R data analysis packages, such as 'caret', a variety of parameter options to help facilitate search automation, a high degree of transparency to the end-user - all intermediate data sets and parameters are made available for further analysis and useful, out-of-the-box visualizations of model performance. Methods for transforming survival data into discrete-time are adapted from the 'autosurv' package by Suresh et al., (2022) <doi:10.1186/s12874-022-01679-6>.

License GPL-3

URL https://github.com/dawdawdo/spect

BugReports https://github.com/dawdawdo/spect/issues

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, randomForest, kernlab, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2025-04-08 09:00:02 UTC

Contents

`create_person_period_data` *Generates person-period data for any data set, given the bounds defined by the training set.*

Description

Generates person-period data for any data set, given the bounds defined by the training set.

Usage

```
create_person_period_data(individual_data, bounds)
```

Arguments

`individual_data`	A survival data set.
`bounds`	Output from the ‘generate_bounds‘ function of this package.

Value

A data set consisting of the original ‘individual_data‘ repeated once for each interval defined by the ‘bounds‘ parameter. Each row will be labeled with an id and an interval. The output of this function can be passed to either ‘create_training_data‘ or ‘spect_predict‘ to genreate modeling data or predictions respectively.

Author(s)

Stephen Abrams, `<stephen.abrams@louisville.edu>`

See Also

[generate_bounds()], [spect_predict()], [create_training_data()]

`create_synthetic_data` *Generates a survival data set for synthetic streaming service subscription data. The survival event in this case is a cancellation of the subscription. It is given as a function of household income and average number of hours watched in the prior month. Users can adjust the level of censoring and variance in the data with the supplied parameters or simply call with no parameters for a default distribution of data.*

Description

Generates a survival data set for synthetic streaming service subscription data. The survival event in this case is a cancellation of the subscription. It is given as a function of household income and average number of hours watched in the prior month. Users can adjust the level of censoring and variance in the data with the supplied parameters or simply call with no parameters for a default distribution of data.

Usage

```
create_synthetic_data(
  sample_size = 250,
  minimum_income = 5000,
  median_income = 50000,
  income_variance = 10000,
  min_watchhours = 0,
  max_watchhours = 6,
  censor_percentage = 0,
  min_censor_amount = 0,
  max_censor_amount = 0,
  study_time_in_months = 48,
  perturbation_shift = 0
)
```

Arguments

`sample_size`	optional - size of the sample population to generate
`minimum_income`	optional - minimum household income used to generate the distribution
`median_income`	optional - median household income used to generate the distribution
`income_variance`	optional - variance to use when generating the household income distribution
`min_watchhours`	optional - minimum average number of hours watched used to generate the distribution
`max_watchhours`	optional - minimum average number of hours watched used to generate the distribution
`censor_percentage`	optional - percentage of population to artificially censor

`min_censor_amount`
: optional - Minimum number of months of censoring to apply to the censored population

`max_censor_amount`
: optional - maximum number of months of censoring to apply to the censored population

`study_time_in_months`
: optional - observation horizon in months

`perturbation_shift`
: optional - defines a boundary for the amount to randomly perturb the formulaic result. Zero for no perturbation

Value

A survival data set suitable for modeling using spect_train.

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

Examples

```
data <- create_synthetic_data()
```

`create_training_data` *Generates modeling data from a person-period data set.*

Description

Generates modeling data from a person-period data set.

Usage

```
create_training_data(person_period_data, time_col, event_col, cens)
```

Arguments

`person_period_data`
: A discrete-time data set. Generally, this will be output from the ‘create_person_period_data‘ function.

`time_col`
: A string specifying the name of the column which contains the survival time.

`event_col`
: A string specifying the name of the column which contains the event indicator.

`cens`
: Specifies how to apply censored data. Valid values are "same" - considers censorship to occur in the same interval as the survival time, "prev" - considers censorship to occur in the prior interval, and "half" - considers censorship to occur in the same interval as survival time if the individual survived for at least half of that interval.

Value

A discrete-time data set suitable for training using any binary classifer.

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

See Also

[create_person_period_data()]

`evaluate_model` *Generates evaluation metrics, include time-dependent TPR and FPR rates as well as AUC*

Description

Generates evaluation metrics, include time-dependent TPR and FPR rates as well as AUC

Usage

```
evaluate_model(train_result, prediction_times, plot_roc = TRUE)
```

Arguments

`train_result`	return data object from spect_train
`prediction_times`	a vecotr of times to use for generating TPR and FPR data
`plot_roc`	optional indicator to display the time-dependent ROC curves. The TPR and FPR data will be returned regardless of the value of this parameter.

Value

Evaluation metrics. Also plots the number of requested samples

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

`generate_bounds` *Generates the intervals based on the survival times in the supplied data set using the quantile function.*

Description

Generates the intervals based on the survival times in the supplied data set using the quantile function.

Usage

```
generate_bounds(
  train_data,
  time_col,
  event_col,
  suggested_intervals,
  obs_window
)
```

Arguments

`train_data`	A survival data set containing at least three columns - one which matches the string in the ‘time_col‘ parameter, one which matches the string in the ‘event_col‘ parameter, and at least one covariate column for modeling.
`time_col`	The name of the column in ‘train_data‘ containing survivial time
`event_col`	The name of the column in ‘train_data‘ contaiing the event indicator. Values in this column must be either zero (0) or one (1)
`suggested_intervals`	The number of intervals to create. If the number of events in the data is less than ‘suggested_intervals‘, it is ignored.
`obs_window`	An artificial censoring time. Any observations in ‘train_data‘ beyond this time will be administratively censored.

Value

A list of upper an lower bounds for each generated interval.

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

See Also

[create_person_period_data()]

Examples

```
df <- data.frame(a=c(1,2,3,4,5,6), surv_time=c(1,4,5,6,8,9), event=c(1,1,1,1,0,1))
bounds <- generate_bounds(df, time_col="surv_time", event_col="event",
                          suggested_intervals=3, obs_window=8)
```

`plot_km` *Plots a series of population Kaplan-Meier curves for different thresholds for both the test predictions and the ground truth*

Description

Plots a series of population Kaplan-Meier curves for different thresholds for both the test predictions and the ground truth

Usage

```
plot_km(train_result, prediction_threshold_search_granularity = 0.05)
```

Arguments

`train_result`	return data object from 'spect_train'
`prediction_threshold_search_granularity`	optional number between zero and one which defines the granularity of searching for cumulative probability thresholds. For instance, search a value of 0.05 will search 19 thresholds (0.05, 0.10, ..., 0.95)

Value

Data used to produce the KM curve and the passed granularity parameter. Also plots the KM curves.

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

`plot_survival_curve` *Plots a sample of individual survival curves from the test data set.*

Description

Plots a sample of individual survival curves from the test data set.

Usage

```
plot_survival_curve(train_result, individual_id, curve_type = "both")
```

Arguments

train_result	return data object from ‘spect_train‘
individual_id	identifier of the individual to plot
curve_type	optional specification of the type of curve. Available options are "conditional", which plots the conditional probability of surviving each interval given that the individual survived to the start of that interval, "absolute" which plots the unconditional probability of surviving each interval, and "both", the default value, which plots both curves on the same chart.

Value

None - plots the number of requested samples

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

---

plot_synthetic_data *Simple visualization of synthetic subscription data.*

Description

Simple visualization of synthetic subscription data.

Usage

```
plot_synthetic_data(data)
```

Arguments

data	a data object generated by create_synthetic_data

Value

None - prints synthetic data generated by create_synthetic_data

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

Examples

```
data <- create_synthetic_data()
plot_synthetic_data(data)
```

`spect_predict` *Generates predictions for each individual at each interval defined by the 'train_result' parameter. The interval-level predictions can be combined to generate surivival curves for an individual.*

Description

Generates predictions for each individual at each interval defined by the 'train_result' parameter. The interval-level predictions can be combined to generate surivival curves for an individual.

Usage

```
spect_predict(train_result, new_data)
```

Arguments

`train_result`	- return data object from spect_train
`new_data`	- New data set with the same covariates as the training data set.

Value

predictions by the trained model on a new data set

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

`spect_train` *Generates a trained caret model using the given primary binary classification. Optionally generates a stacked ensemble model if a list of base learners is supplied.*

Description

Generates a trained caret model using the given primary binary classification. Optionally generates a stacked ensemble model if a list of base learners is supplied.

Usage

```
spect_train(
  test_prop = 0.2,
  censor_type = "half",
  bin_slices = 10,
  method = "repeatedcv",
  resampling_number = 10,
```

```
  kfold_repeats = 3,
  model_algorithm,
  base_learner_list = list(),
  metric = "Kappa",
  rng_seed = 42,
  use_parallel = TRUE,
  cores = 0,
  modeling_data,
  event_indicator_var,
  survival_time_var,
  obs_window
)
```

Arguments

test_prop	optional proportion of the data set to reserve for testing
censor_type	optional method used to determine censorship in a given bin - may be "half", "prev" or "same". see createDiscreteDat for usage.
bin_slices	optional number of intervals to use for predictions.
method	optional caret parameter
resampling_number	optional for repeated cv
kfold_repeats	optional number of folds
model_algorithm	primary classification algorithm. Trains a stack-ensemble model if 'base_learner_list' is supplied, otherwise trains a simple classifier model.
base_learner_list	optional list of base learner algorithms
metric	optional metric for model calibration
rng_seed	optional random number generation seed for reproducibility
use_parallel	optioanlly make use of the caret multicore training cluster
cores	optioanl number of cores for multicore training. If zero, spect will attempt to make a good choice. Note: only relevant if 'use_parallel' is set to TRUE, otherwise this parameter is ignored.
modeling_data	This data set must have one column for time and one column for the event indicator. The remaining columns are treated as covariates for modeling.
event_indicator_var	The name of the column containing the event indicator (values in this column must be zero or one).
survival_time_var	The name of the column containing the time variable
obs_window	The last time to use for generating person-period data. Any event occurring after this time will be administratively censored. In general, choosing a time at or near the end of the max observed time will include most events.

Value

A list containing all intermediate data sets created by ‘spect_train‘, a trained caret model object, the following parameters passed to ‘spect_train‘: ‘obs_window‘, ‘survival_time_var‘, ‘event_indicator_var‘, ‘base_learner_list‘, ‘bin_slices‘, and the bounds of each interval generated by the training data set.

Author(s)

Stephen Abrams, <stephen.abrams@louisville.edu>

Index