Package 'SURVFIT'

December 16, 2020

Type Package

Title Rule Ensemble Learning and Analysis for Survival Data

Version 0.1.0

Author@R
c(person(``Ameer'',``Hamza''), email = <ameerh.shakur@gmail.com>, role = c(``aut'',``cre''))

Author Ameer Hamza [aut, cre]

Maintainer Ameer Hamza <ameerh.shakur@gmail.com>

Description This package derives doubly sparse predictive
ensemble of rules from survival data using a second order cone
formulation solved using Cplex. Double sparsity induces sparsity
in both number of rules extracted as well as variables involved
in the rules. This package also includes methods for comprehensive
analysis of extracted rules such asstatistical testing,
decomposition analysis and sensitivity analysis.

URL <https://github.com/hamzameer/SURVFIT>

Depends R (>= 3.5.3)

License GPL-2 | GPL-3

Encoding UTF-8

LazyData true

Imports osqp,
dplyr,
ranger,
survival,
ggplot2,
Formula,
Rcplex

RoxygenNote 7.0.2

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

1

| hello | *Hello, World!* |

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

| SURVFIT | *Doubly Sparse Survival Rule Extraction* |

Description

SURVFIT extracts a "doubly sparse" (sparse in both number of rules and in number of variables in the rules) survival rule ensemble from survival data

Usage

```
SURVFIT(
  formula = formula,
  data = data,
  rulelength = 3,
  doubly.sparse = FALSE,
  gamma = NULL,
  lambda1 = NULL,
  lambda2 = NULL,
  crossvalidate = TRUE,
  nfolds = 4,
  num_toprules = 16,
  num_totalrules = 2000,
  input_rule_list = FALSE,
  rule_list = NULL,
  ntree = 200,
  digit = 10,
  seed = NULL,
  nodesize = NULL,
  trace = 1,
  max.grid = 25,
  ...
)
```

Arguments

formula	`formula`. The model formula specifying time, status and dependent variables of the form `Surv(time, status)~ x1 + x2 + ..`
data	`data.frame`. Training data.
rulelength	`Integer`. Maximum length of the rule. (Default = 3)
doubly.sparse	`Logical` for whether double sparsity required. (Default = FALSE)
gamma	`Numeric` or `list`. Hyperparameter (Default = NULL)
lambda1	`Numeric` or `list`. Hyperparameter (Default = NULL)
lambda2	`Numeric` or `list`. Hyperparameter (Default = NULL)
crossvalidate	`Logical`. Whether crossvalidation to be done to find hyperparameters. (Default = TRUE)
nfolds	`Integer`. Number of cross validation folds. (Default = 5)
num_toprules	`Integer`. Number of rules extracted. (Default = 16)
num_totalrules	`Integer`. Number of rules considered. (Default = 2000)
input_rule_list	`Logical` Whether rule list supplied. (Default = FALSE)
rule_list	`List`. List of supplied rules. (Default = NULL)
ntree	`Integer` .Number of trees built
digit	`Integer`. Decimal points.
seed	`Numeric`. Seed for reproducible experiments.
nodesize	`Integer`. (Default = NULL)
trace	`0` or `1`. : Turn CPLEX output on (1) or off(0). Default 1.
...	Other inputs

Value

Object of class `list` with elements

rules	List of top num_toprules rules
all_rules	List of all num_totalrules rules
rule_data	`Data.frame` of rules evaluated over data
beta	Coefficients of all_rules in the model

Examples

```
## For ovarian data from "survival" package.
SURVFIT(Surv(futime, fustat) ~ ., data = ovarian)
```

Index