Package ‘iBST’

April 18, 2022

Type Package
Title Improper Bagging Survival Tree
Version 1.1
Date 2022-04-17
Author Cyprien Mbogning and Philippe Broet
Maintainer Cyprien Mbogning <cyprien.mbagning@gmail.com>
Description Fit a bagging survival tree on a mixture of population (susceptible and nonsusceptible) using either a pseudo R2 criterion or an adjusted Logrank criterion. The predictor is evaluated using the Out Of Bag Integrated Brier Score (IBS) and several scores of importance are computed for variable selection. The thresholds values for variable selection are computed using a nonparametric permutation test.
License GPL (>= 2)
LazyLoad yes
Depends survival , rpart , parallel
Imports Rcpp (>= 1.0.8)
LinkingTo Rcpp
NeedsCompilation yes
Repository CRAN
Date/Publication 2022-04-18 11:12:31 UTC

R topics documented:

ibST-package ................................................. 2
Bagg_pred_Surv ............................................. 4
Bagg_Surv .................................................... 6
burn ............................................................. 9
improper_tree .............................................. 10
permute_select_surv ....................................... 11
PseudoR2.Cure ............................................ 13
cpp_hello_world .......................................... 14
tree2indicators ............................................ 15
Description

Fit a bagging survival tree on a mixture of population (susceptible and nonsusceptible) using either a pseudo R2 criterion or an adjusted Logrank criterion. The predictor is evaluated using the Out Of Bag Integrated Brier Score (IBS) and several scores of importance are computed for variable selection. The thresholds values for variable selection are computed using a nonparametric permutation test. See Cyprien Mbogning and Philippe Broet (2016)<doi:10.1186/s12859-016-1090-x> for an overview about the methods implemented in this package.

Details

Package: iBST
Type: Package
Version: 1.1
Date: 2022-04-17
License: GPL(>=2.0)

Author(s)

Cyprien Mbogning and Philippe Broet
Maintainer: Cyprien Mbogning <cyprien.mbozign@gmail.com>

References


See Also

Bagg_Surv Bagg_pred_Surv improper_tree

Examples

## Not run:
data(burn)
myarg = list(cp = 0, maxcompete = 0, maxsurrogate = 0, maxdepth = 2)
Y.names = c("T3", "D3")
P.names = 'Z2'
T.names = c("Z1", paste("Z", 3:11, sep = ''))
mybag = 40
set.seed(5000)

## fit an improper survival tree
burn.tree <- suppressWarnings(improper_tree(burn,
    Y.names, P.names, T.names, method = "R2",
    args.rpart = myarg))

plot(burn.tree)
text(burn.tree, cex = .7, xpd = TRUE)

## fit an improper Bagging survival tree with the adjusted Logrank criterion
burn.BagEssai0 <- suppressWarnings(Bagg_Surv(burn,
    Y.names, P.names, T.names, method = "LR",
    args.rpart = myarg,
    args.parallel = list(numWorkers = 1),
    Bag = mybag))

## fit an improper Bagging survival tree with the pseudo R2 criterion
burn.BagEssai1 <- suppressWarnings(Bagg_Surv(burn,
    Y.names, P.names, T.names, method = "R2",
    args.rpart = myarg,
    args.parallel = list(numWorkers = 1),
    Bag = mybag))

## Plot the variable importance scores
par(mfrow=c(1,3))
barplot(burn.BagEssai1$IIS,
    main = 'IIS', horiz = TRUE, las = 1,
    cex.names = .8, col = 'lightblue')

barplot(burn.BagEssai1$DIIS,
    main = 'DIIS', horiz = TRUE, las = 1,
    cex.names = .8, col = 'grey')

barplot(burn.BagEssai1$DEPTH,
    main = 'MinDepth', horiz = TRUE, las = 1,
Bagging survival tree prediction

Description

Use the Bagging improper survival tree to predict on new features and to evaluate the predictor using Out Of Bag Integrated Brier Scores with either the Nelson Aalen estimator or the Breslow estimator. A permutation importance score is also computed using OOB observations.

Usage

Bagg_pred_Surv(xdata, Y.names, P.names, resBag, args.parallel = list(numWorkers = 1),
   new_data = data.frame(), OOB = FALSE)

Arguments

xdata The learning data frame
Y.names A vector of the names of the two variables of interest (the time-to-event is follow by the event indicator)
P.names The names of independant variables acting on the non-susceptible population (the plateau)
resBag The result of the Bagg_Surv function
args.parallel a list containing the number of parallel computing arguments: The number of workers, the type of parallelization to achieve, ... see mclapply for further details.
**Bagg_pred_Surv**

**new_data**
An optional data frame to validate the bagging procedure (the test dataset)

**OOB**
A value of TRUE or FALSE with TRUE indicating the computation of the OOB error using the Integrated Brier Score and also the computation of the permutation importance score.

**Value**

**PREDNA**
A matrix with Nelson Aalen predictions on all individuals of the learning sample

**PREDBRE**
A matrix with Breslow predictions on all individuals of the learning sample

**tabhazNAa**
A list of matrix with Nelson Aalen prediction of each tree of the bagging sequence with the leaf node prediction in each column

**tabhazBRe**
A list of matrix with Breslow prediction of each tree of the bagging sequence with the leaf node prediction in each column

**OOB**
A value of NULL if OOB is FALSE. A list of twelve elements otherwise: IBSKM: The Kaplan-Meier estimation of the Integrated Brier Score; IBSNAOOB: The OOB error using the Nelson-Aalen estimator; IBSBREOOB: The OOB error using the Breslow estimator; vimpooobpbpna: The permutation variable importance using the Nelson-Aalen estimator; vimpooobpbpbre: The permutation variable importance using the Breslow estimator; oobibspbpna: The mean OOB error predictor by predictor using the Nelson-Aalen estimator; oobibspbpbre: The mean OOB error predictor by predictor using the Breslow estimator; SURVNAOOB: A matrix with the predicted OOB survival using the Nelson-Aalen estimator; SURVBREOOB: A matrix with the predicted OOB survival using the Breslow estimator; BSTKM: The vector of Brier scores using the KM estimator; BSTNAOOB: The vector of Brier scores using the NA estimator; BSTBREOOB: The vector of Brier scores using the BRE estimator.

**Timediff**
The execution time of the prediction procedure

**TEST**
A value of NULL if new_data is not available. A list of seven elements otherwise: IBSNAKMnew: The IBS using the NA estimator on the new dataset; IBSBRKMnew: The IBS using the BRE estimator on the new dataset; IBSSKnew: The IBS using the KM estimator on the new dataset; SURVNAnew: A matrix of predicted survival on the new dataset using the NA estimator; SURVBREnew: A matrix of predicted survival on the new dataset using the BRE estimator; SURV_NAnew: a vector of survival prediction on the testing dataset using the NA estimator; SURV_BREnew: a vector of survival prediction on the testing dataset using the BRE estimator.

**Author(s)**
Cyprien Mbogning and Philippe Broet

**References**

**See Also**

Bagg_Surv
Examples

```r
## Not run:
data(burn)
myarg = list(cp = 0, maxcompete = 0, maxsurrogate = 0, maxdepth = 2)
Y.names = c("T3", "D3")
P.names = c("Z2")
T.names = c("Z1", paste("Z", 3:11, sep = ""))
mybag = 40
set.seed(5000)
burn.BagEssai0 <- Bagg_Surv(burn, Y.names, P.names, T.names, method = "LR", args.rpart = myarg,
                           args.parallel = list(numWorkers = 1), Bag = mybag)
burn.BagEssai1 <- Bagg_Surv(burn, Y.names, P.names, T.names, method = "R2", args.rpart = myarg,
                           args.parallel = list(numWorkers = 1), Bag = mybag)
pred0 <- Bagg_pred_Surv(burn, Y.names, P.names, burn.BagEssai0,
                        args.parallel = list(numWorkers = 1), OOB = TRUE)
pred1 <- Bagg_pred_Surv(burn, Y.names, P.names, burn.BagEssai1,
                        args.parallel = list(numWorkers = 1), OOB = TRUE)
## End(Not run)
```

Bagg_Surv

Bagging improper survival trees

Description

Bagging procedure to aggregate several improper trees using either the pseudo-R2 procedure or the adjusted Logrank procedure. Several scores for variables importance are computed.

Usage

```r
Bagg_Surv(xdata,
          Y.names,
          P.names,
          T.names,
          method = "R2",
          args.rpart,
```
args.parallel = list(numWorkers = 1),
Bag = 100)

Arguments

xdata: The learning data frame
Y.names: A vector of the names of the two variables of interest (the time-to-event is followed by the event indicator)
P.names: The names of independent variables acting on the non-susceptible population (the plateau)
T.names: The names of independent variables acting on the survival of the susceptible population
method: The chosen method (either "LR" for the Logrank or "R2" for the proposed pseudo-R2 criterion)
args.rpart: The improper survival tree parameters: a list of options that control details of the rpart algorithm. minbucket: the minimum number of observations in a terminal <leaf> node; cp: complexity parameter (Any split that does not decrease the overall lack of fit by a factor of cp is not attempted); maxdepth: the maximum depth of any node of the final tree, with the root node counted as depth 0. ... See rpart.control for further details.
args.parallel: a list containing the number of parallel computing arguments: The number of workers, the type of parallelization to achieve, ... see mclapply for further details.
Bag: The number of Bagging samples to consider

Details

For the Bagging procedure, it is mandatory to set maxcompete = 0 and maxsurrogate = 0 within the args.rpart arguments. This will ensure the correct calculation of the importance of variables and also a better computation time.

Value

A list of ten elements

MaxTreeList: The list of improper survival trees computed during the bagging procedure
IIS: The Index Importance Score
DIIS: The Depth Index Importance Score
DEPTH: The minimum depth importance Score
IND_OOB: A list of length Bag containing the Out Of Bag (OOB) individuals for improper survival tree model
IIND_SAMP: The final list of length Bag of sample individuals used for each improper survival tree
IIND_SAMP: The initial list of sample individuals used for each improper survival tree at the beginning
Bagg_Surv

Bag  The number of bagging samples retained at the end of the procedure after removing the trees without leaves
indrpart  a vector of TRUE or FALSE with the value FALSE when the corresponding tree is removed from the final bagged predictor
Timediff  The elapsed time of the Bagging procedure

Note

This version of the code allows for the moment only one variable to have an impact on the cured population. The next version will allow more than one variable.

Author(s)

Cyprien Mbogning and Philippe Broet

References


See Also

Bagg_pred_Surv

Examples

```r
## Not run:
data(burn)
myarg = list(cp = 0, maxcompete = 0, maxsurrogate = 0, maxdepth = 2)
Y.names = c("T3", "D3")
P.names = 'Z2'
T.names = c("Z1", paste("Z", 3:11, sep = ' '))
mybag = 40
set.seed(5000)
burn.BagEssai0 <- Bagg_Surv(burn,
  Y.names,
P.names,
  T.names,
  method = "LR",
  args.rpart = myarg,
  args.parallel = list(numWorkers = 1),
  Bag = mybag)
burn.BagEssai1 <- Bagg_Surv(burn,
  Y.names,
P.names,
  T.names,
  method = "R2",
  args.rpart = myarg,
  args.parallel = list(numWorkers = 1),
```
burn

Bag = mybag)

## End(Not run)

---

**burn**  
**burn dataset**

---

### Description

The burn data frame has 154 rows and 17 columns.

### Usage

data(burn)

### Format

A data frame with 154 observations on the following 17 variables.

- **Obs**: Observation number
- **Z1**: Treatment: 0=routine bathing 1=Body cleansing
- **Z2**: Gender (0=male 1=female)
- **Z3**: Race: 0=nonwhite 1=white
- **Z4**: Percentage of total surface area burned
- **Z5**: Burn site indicator: head 1=yes, 0=no
- **Z6**: Burn site indicator: buttock 1=yes, 0=no
- **Z7**: Burn site indicator: trunk 1=yes, 0=no
- **Z8**: Burn site indicator: upper leg 1=yes, 0=no
- **Z9**: Burn site indicator: lower leg 1=yes, 0=no
- **Z10**: Burn site indicator: respiratory tract 1=yes, 0=no
- **Z11**: Type of burn: 1=chemical, 2=scald, 3=electric, 4=flame
- **T1**: Time to excision or on study time
- **D1**: Excision indicator: 1=yes 0=no
- **T2**: Time to prophylactic antibiotic treatment or on study time
- **D2**: Prophylactic antibiotic treatment: 1=yes 0=no
- **T3**: Time to staphylococcal aureus infection or on study time
- **D3**: Staphylococcal aureus infection: 1=yes 0=no

### Source

improper_tree

**Examples**

```r
data(burn)
## maybe str(burn)
```

---

**improper_tree**  
*improper survival tree*

---

**Description**

Fit an improper survival tree for the mixed population (susceptible and nonsusceptible) using either the proposed pseudo R2 criterion or an adjusted Logrank criterion.

**Usage**

```r
improper_tree(xdata,  
Y.names,  
P.names,  
T.names,  
method = "R2",  
args.rpart)
```

**Arguments**

- `xdata`: The learning data frame
- `Y.names`: A vector of the names of the two variables of interest (the time-to-event is followed by the event indicator)
- `P.names`: The names of independent variables acting on the non-susceptible population (the plateau)
- `T.names`: The names of independent variables acting on the survival of the susceptible population
- `method`: The chosen method (either "LR" for the Logrank or "R2" for the proposed pseudo-R2 criterion)
- `args.rpart`: The improper survival tree parameters: a list of options that control details of the rpart algorithm. `minbucket`: the minimum number of observations in any terminal <leaf> node; `cp`: complexity parameter (Any split that does not decrease the overall lack of fit by a factor of `cp` is not attempted); `maxdepth`: the maximum depth of any node of the final tree, with the root node counted as depth 0. ... See `rpart.control` for further details.

**Value**

An unpruned improper survival tree.

**Author(s)**

Cyprien Mbogning and Philippe Broet
References


See Also

Bagg_Surv Bagg_pred_Surv

Examples

```r
## Not run:
data(burn)
myarg = list(cp = 0, maxcompete = 0, maxsurrogate = 0, maxdepth = 3)
Y.names = c("T3", "D3")
P.names = 'Z2'
T.names = c("Z1", paste("Z", 3:11, sep = ''))
burn.tree <- suppressWarnings(improper_tree(burn, Y.names, P.names, T.names, method = "R2", args.rpart = myarg))

plot(burn.tree)
text(burn.tree, cex = .7, xpd = TRUE)
## End(Not run)
```

permute_select_surv

permutation variable selection

Description

Variable selection using the permutation test on several scores of importance: IIS, DIIS and DEPTH.

Usage

```r
permute_select_surv(xdata, Y.names, P.names, T.names, importance = "IIS", method = "R2", Bag, args.rpart, args.parallel = list(numWorkers = 1), nperm = 50)
```
Arguments

xdata  The learning data frame
Y.names  A vector of the names of the two variables of interest (the time-to-event is follow
         by the event indicator)
P.names  The names of independent variables acting on the non-susceptible population
         (the plateau)
T.names  The names of independent variables acting on the survival of the susceptible
         population
importance  The importance score to consider: either IIS, DIIS or DEPTH
method  The splitting method: either "R2" for the proposed pseudo-R2 criterion or "LR"
         for the adjusted Logrank criterion
Bag  The number of Bagging samples to consider
args.rpart  The improper survival tree parameters: a list of options that control details of the
            rpart algorithm. minbucket: the minimum number of observations in any terminal
            <leaf> node; cp: complexity parameter (Any split that does not decrease the
            overall lack of fit by a factor of cp is not attempted); maxdepth: the maximum
            depth of any node of the final tree, with the root node counted as depth 0. ... See
            rpart.control for further details
args.parallel  a list containing the number of parallel computing arguments: The number of
               workers, the type of parallelization to achieve, ... see mclapply for further de-
               tails.
nperm  The number of permutation samples to consider for the permutation test

Details

Testing whether the importance score is null or not.

Value

A list of five elements:

- pvalperm1  The permutation test P-values ranking in decreasing order
- pvalperm2  The permutation test P-values ranking in decreasing order considering an ap-
             proximate gaussian distribution under the null hypothesis
- pvalKS  The Kolmogorov-Smirnov P-values of the comparisons between the observed
          importance under the null hypothesis and a theoretical gaussian distribution
- IMPH1  The observed importance score
- PERMH0  A matrix with the importance scores for each permutation sample in each col-

Author(s)

Cyprien Mbogning and Philippe Broet
References

See Also
Bagg_Surv Bagg_pred_Surv

Examples
## Not run:
myarg = list(cp = 0, maxcompete = 0, maxsurrogate = 0, maxdepth = 2)
Y.names = c("T3", "D3")
P.names = 'Z2'
T.names = c("Z1", paste("Z", 3:11, sep = ''))
mybag = 40
set.seed(5000)
data(burn)
resperm0 <- suppressWarnings(permute_select_surv(xdata = burn,
    Y.names, P.names, T.names,
    method = "LR",
    Bag = mybag,
    args.rpart = myarg,
    args.parallel = list(numWorkers = 1),
    nperm = 150))

## End(Not run)

PseudoR2.Cure

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pseudo R2 criterion for a mixture of population (susceptible and nonsusceptible populations)</td>
</tr>
</tbody>
</table>

Usage

PseudoR2.Cure(ygene, ydelai, yetat, strate, ordered = FALSE)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ygene</td>
<td>The main variable of interest</td>
</tr>
<tr>
<td>ydelai</td>
<td>The right censored delay until the event</td>
</tr>
<tr>
<td>yetat</td>
<td>The censoring indicator</td>
</tr>
<tr>
<td>strate</td>
<td>The variables acting on the nonsusceptible or cured population</td>
</tr>
<tr>
<td>ordered</td>
<td>A value of TRUE or FALSE indicating whether or not the times to event are ordered</td>
</tr>
</tbody>
</table>
rcpp_hello_world

Value

A pseudo R2 value lying between 0 and 1.

Author(s)

Cyprien Mbogning and Philippe Broet

References


See Also

Bagg_Surv Bagg_pred_Surv improper_tree

Examples

data(burn)
PseudoR2.Cure(ygene = burn$Z3,
    ydelay = burn$delay,
    yetat = burn$yetat,
    strate = burn$strate)

PseudoR2.Cure(ygene = burn$Z2,
    ydelay = burn$delay,
    yetat = burn$yetat,
    strate = burn$strate)

rcpp_hello_world

Simple function using Rcpp

Description

Simple function using Rcpp

Usage

rcpp_hello_world()

Examples

## Not run:
rcpp_hello_world()

## End(Not run)
**tree2indicators**

*From a tree to indicators (or dummy variables)*

---

**Description**

Coerces a given tree structure inheriting from rpart to binary covariates.

**Usage**

```r
tree2indicators(fit)
```

**Arguments**

- `fit` a tree structure inheriting to the rpart method

**Value**

A list of indicators defining the leaf nodes of the fitted tree from left to right.

**Author(s)**

Cyprien Mbogning

**Examples**

```r
fit <- rpart(Kyphosis ~ Age + Number + Start, data = kyphosis)
tree2indicators(fit)
```
Index

* datasets
  burn, 9
* documentation
  Bagg_pred_Surv, 4
  Bagg_Surv, 6
  iBST-package, 2
  improper_tree, 10
  permute_select_surv, 11
  PseudoR2.Cure, 13
  tree2indicators, 15
* multivariate
  Bagg_pred_Surv, 4
  Bagg_Surv, 6
  iBST-package, 2
  improper_tree, 10
  permute_select_surv, 11
* nonparametric
  permute_select_surv, 11
* package
  iBST-package, 2
* survival
  Bagg_pred_Surv, 4
  Bagg_Surv, 6
  iBST-package, 2
  improper_tree, 10
  permute_select_surv, 11
  PseudoR2.Cure, 13
* tree
  Bagg_pred_Surv, 4
  Bagg_Surv, 6
  iBST-package, 2
  improper_tree, 10
  permute_select_surv, 11
  PseudoR2.Cure, 13
  tree2indicators, 15

Bagg_pred_Surv, 2, 4, 8, 11, 13, 14
Bagg_Surv, 2, 4, 5, 6, 11, 13, 14
burn, 9