Package 'OSTE'

October 12, 2022

Type Package

Title Optimal Survival Trees Ensemble

Version 1.0

Date 2021-11-07

Maintainer Naz Gul <nazgul@awkum.edu.pk>

Description

Function for growing survival trees ensemble ('Naz Gul', 'Nosheen Faiz', 'Dan Brawn', 'Rafal Kulakowski', 'Zardad Khan', and 'Berthold Lausen' (2020) <arXiv:2005.09043>) is given.
The trees are grown by the method of random survival forest ('Marvin Wright', 'Andreas Ziegler' (2017) <doi:10.18637/jss.v077.i01>).
The survival trees grown are assessed for both individual and collective performances.
The ensemble can give promising results on fewer survival trees selected in the final ensemble.

Imports ranger, pec, stats, survival, prodlim

LazyLoad yes

License GPL (>= 3.5.0)

NeedsCompilation no

Author Naz Gul [aut, cre],
Nosheen Faiz [ctb],
Zardad Khan [aut],
Berthold Lausen [aut]

Repository CRAN

Date/Publication 2021-11-19 14:50:09 UTC

R topics documented:

OSTE-package   *Optimal Survival Trees Ensembles*

Description

This package consists of function for growing survival trees ensemble, that are grown by the method of random survival forest. The survival trees grown are assessed for both individual and collective performances. The ensemble can give promising results on fewer survival trees selected based on their individual and collective performance in the final ensemble.

Details

Package:	OSTE
Type:	Package
Version:	1.0
Date:	2021-11-07
License:	GPL (>= 3.5.0)

Author(s)

Naz Gul, Nosheen Faiz, Zardad Khan and Berthold Lausen.

Maintainer: Naz Gul <nazgul@awkum.edu.pk>

References

Gul, N., Faiz, N., Brawn, D., Kulakowski, R., Khan, Z., & Lausen, B. (2020). Optimal survival trees ensemble. arXiv preprint arXiv:2005.09043.

OSTE   *Optimal Survival Tree Ensemble*

Description

Optimal survival trees ensemble is the main function of `OSTE` package that grows a sufficiently large number, `t.initial`, of survival trees and selects optimal survival trees from the total trees grown by random survival forest. Number of survival trees in the initial set, `t.initial`, is chosen by the user. If not chosen, then the default `t.initial = 500` is used. Based on empirical investigation, `t.initial =1000` is recommended.

Usage

```
OSTE(formula = NULL, data, t.initial = NULL, v.size = NULL, mtry = NULL, M = NULL,
minimum.node.size = NULL, always.split.features = NULL, replace = TRUE,
splitting.rule = NULL, info = TRUE)
```

Arguments

formula	Object of class formula describing the required model to be fitted. Interaction terms are not supported in the current version.
data	A nxd matrix or data frame of n observations on d features along with response variables that are described by the formula.
t.initial	Number of survival trees to be grown initially. If equal to NULL then the defalut of t.initial = 500 is taken. A recommended value is t.initial = 1000.
v.size	Portion of data used for validation in the second phase i.e. for assessing survival trees performance in the ensemble. If equal to NULL then the defalut v.size=0.1
mtry	Number of features selected at random at each node of the survival trees for splitting. If equal to NULL then the default sqrt(d) is taken.
M	Percent of the best t.initial survival trees to be selected on the basis of their performance on out-of-bag observations. For selecting 20% of trees, take M=0.2.
minimum.node.size	
	Minimal node size. If equal to NULL then the default minimum.node.size = 3 is executed.
always.split.features	
	Vector of variable names if desired to be always selected in addition to the mtry variables tried for splitting.
replace	Whether sampling should be done with or without replacement.
splitting.rule	Splitting rule."logrank", "C" or "maxstat" are suported with default "logrank".
info	If TRUE, displays process status .

Details

Large values are recommended for t.initial for better performance as possible under the available computational resources. The log-rank test statistic is used as defalut, A C-index based splitting rule (Schmid et al. 2015) and maximally selected rank statistics (Wright et al. 2016) are available. The C-index shows better predictive performance in case of high censoring rate, where logrank is best for situations where the data are noisy (Schmid et al. 2015).

Value

unique.death.times	
	Unique death times.
CHF	Estimated cumulative hazard function for each observation.
Survival_Prob	Estimated survival probability for each observation.
trees_selected	Number of trees selected.
mtry	Value of mtry used.
forest	Saved forest for prediction purposes.

Note

In the case of missing values in any dataset prior action needs to be taken as the fuction can not handle them at the current version. Moreover, the status/delta variable in the data must be code as 0, 1.

Author(s)

Naz Gul, Nosheen Faiz, Zardad Khan and Berthold Lausen.

References

Marvin N. Wright, Andreas Ziegler (2017). ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. Journal of Statistical Software, 77(1), 1-17. doi:10.18637/jss.v077.i01

Terry Therneau, Beth Atkinson and Brian Ripley (2015) rpart: Recursive Partitioning and Regression Trees. R package version 4.1-10. https://CRAN.R-project.org/package=rpart

Ulla B. Mogensen, Hemant Ishwaran, Thomas A. Gerds (2012). Evaluating Random Forests for Survival Analysis Using Prediction Error Curves. Journal of Statistical Software, 50(11), 1-23. URL http://www.jstatsoft.org/v50/i11/.

Schmid, M., Wright, M. N. & Ziegler, A. (2016). On the use of Harrell's C for clinical risk prediction via random survival forests. Expert Syst Appl 63:450-459. http://dx.doi.org/10.1016/j.eswa.2016.07.018.

Wright, M. N., Dankowski, T. & Ziegler, A. (2017). Unbiased split variable selection for random survival forests using maximally selected rank statistics. Stat Med. http://dx.doi.org/10.1002/sim.7212.

Zardad Khan, Asma Gul, Aris Perperoglou, Osama Mahmoud, Werner Adler, Miftahuddin and Berthold Lausen (2015). OTE: Optimal Trees Ensembles for Regression, Classification and Class Membership Probability Estimation. R package version 1.0. https://CRAN.R-project.org/package=OTE

Gul, N., Faiz, N., Brawn, D., Kulakowski, R., Khan, Z., & Lausen, B. (2020). Optimal survival trees ensemble. arXiv preprint arXiv:2005.09043.

See Also

VETERAN

Examples

```
#Load the data
data(VETERAN)
library(survival)
library(prodlim)
library(ranger)
library(pec)
#Divide the data into training and test parts


 predictSurvProb.ranger <- function (object, newdata, times, ...) {

   ptemp <- ranger:::predict.ranger(object, data = newdata, importance = "none")$survival
```

```
      pos <- sindex(jump.times = object$unique.death.times,
                           eval.times = times)
      p <- cbind(1, ptemp)[, pos + 1, drop = FALSE]
      if (NROW(p) != NROW(newdata) || NCOL(p) != length(times))
       stop(paste("\nPrediction matrix has wrong dimensions:\nRequested newdata x times: ",
                  NROW(dts[trainind,]), " x ", length(1), "\nProvided prediction matrix: ",
                    NROW(p), " x ", NCOL(p), "\n\n", sep = ""))
      p
   }

n <- nrow(VETERAN)
trainind <- sample(1:n,n*0.7)
testind <- (1:n)[-trainind]

# Grow OSTE on the training data

OSTE.fit <- OSTE(Surv(time,status)~.,data=VETERAN[trainind,],t.initial=100)

# Predict on the test data

pred <- ranger:::predict.ranger(OSTE.fit$forest,data=VETERAN[testind,])

# Index various values

pred$survival
pred$survival

#etc.

# To calculate IBS
# Create formula
frm <- as.formula(Surv(time, status) ~ trt + celltype + karno + diagtime + age + prior)

PredError <- pec(object=OSTE.fit$forest, exact==TRUE,
                 formula = frm, cens.model="marginal",
                 data=VETERAN[testind,], verbose=F)
IBS <- crps(object = PredError, times =100, start = PredError$start)[2,1]
IBS
```

| VETERAN | *Data on randomized trial of two treatment procedures for lung cancer.* |

Description

The data set consist of a total 137 observations on 8 variables. The variables consist of the type of lung cancer treatment i.e 1 (standard) and 2 (test drug), cell Type, Status, that denotes the status of the patient as 1 (dead) or 0 (alive), survival time in days since the treatment, Diag, the time since diagnosis in months, age in years, the Karnofsky score, therapy that denotes any prior therapy 0 (none), 1 (yes).

Usage

```
data("VETERAN")
```

Format

A data frame with 137 observations on the following 8 variables.

trt a numeric vector denoting type of lung cancer treatment i.e 1 (standard) and 2 (test drug).

celltype a factor with levels squamous, smallcell, adeno and large.

time a numeric vector denoting survival time in days since the treatment.

status a numeric vector that denotes the status of the patient as 1 (dead) or 0 (alive).

karno a numeric vector denoting the Karnofsky score.

diagtime a numeric vector denoting the time since diagnosis in months.

age age in years.

prior a numeric vector denoting prior therapy; 0 (none), 1 (yes).

References

Therneau T (2015). A Package for Survival Analysis in S. version 2.38, <URL: https://CRAN.R-project.org/package=survival>.

Terry M. Therneau and Patricia M. Grambsch (2000). Modeling Survival Data: Extending the Cox Model. Springer, New York. ISBN 0-387-98784-3

Examples

```
#To load the data
data(VETERAN)
# To see the structure
str(VETERAN)
#etc.
```

Index