

Package ‘dtree’

April 20, 2017

Type Package

Title Decision Trees

Version 0.2.3

Author Ross Jacobucci

Maintainer Ross Jacobucci <rcjacobuc@gmail.com>

Description Combines various decision tree algorithms, plus both linear regression and ensemble methods into one package. Allows for the use of both continuous and categorical outcomes. An optional feature is to quantify the (in)stability to the decision tree methods, indicating when results can be trusted and when ensemble methods may be preferential.

License GPL (>= 2)

LazyData TRUE

RoxygenNote 6.0.1

Depends rpart, party, evtree, partykit, caret

Suggests randomForest, tree, MASS, ISLR, matrixStats, plyr, rpart.utils, stringr, pROC

NeedsCompilation no

Repository CRAN

Date/Publication 2017-04-20 15:06:11 UTC

R topics documented:

dtree	2
stable	3
summary.dtree	4

Index	5
--------------	----------

dtree

*Main function for creating different types of decision trees***Description**

Main function for creating different types of decision trees

Usage

```
dtree(formula, data, methods = c("lm", "rpart", "tree", "ctree", "evtree"),
      samp.method = "repeatedcv", tuneLength = 3, bump.rep = 50,
      subset = FALSE, perc.sub = 0.75, weights = NULL, verbose = TRUE)
```

Arguments

formula	a formula, with a response to left of ~.
data	Data frame to run models on
methods	Which tree methods to use. Defaults: lm, rpart, ctree, evtree. Also can use "rf" for random forests. Also a FDR pruning method for ctree termed "ctreePrune". Finally bumping is implemented as methods="bump".
samp.method	Sampling method. Refer to caret package trainControl() documentation. Default is repeated cross-validation. Other options include "cv" and "boot".
tuneLength	Number of tuning parameters to try. Applies to train(). Can also be specified as a vector, with order corresponding to the order specified in the methods argument.
bump.rep	Number of repetitions for bumping
subset	Whether to split dataset into training and test sets
perc.sub	What fraction of data to put into train dataset. 1-frac.sub is allocated to test dataset. Defaults to 0.75
weights	Optional weights for each case.
verbose	Whether to print what method on

Examples

```
# continuous outcome
#library(MASS) # for boston data
#data(Boston)
#out <- dtree(medv ~ ., data=Boston,methods=c("lm","rpart","ctree"))
#summary(out)
# plot(out$rpart.out)

# categorical outcome
#library(ISLR)
#data(Default)

#out <- dtree(default ~ ., data=Default,methods=c("lm","rpart"))
#summary(out)
```

stable	<i>Main function to calculate stability coefficients</i>
--------	--

Description

Main function to calculate stability coefficients

Usage

```
stable(formula, data, methods = c("lm", "rpart", "tree", "ctree", "evtree"),
  samp.method = "repeatedcv", tuneLength = 3, n.rep = 100,
  bump.rep = 50, parallel = FALSE, ncore = detectCores() - 1,
  roundVal = 1, stablelearner = FALSE, subset = FALSE, perc.sub = 0.75,
  weights = NULL)
```

Arguments

formula	a formula, weight a response to left of ~.
data	Data frame to run models on
methods	Which tree methods to use. Defaults: lm, rpart, tree, ctree, evtree. Also can use "rf" for random forests
samp.method	Sampling method. Refer to caret package trainControl() documentation. Default is repeated cross-validation. Other options include "cv" and "boot".
tuneLength	Number of tuning parameters to try. Applies to train()
n.rep	Number of times to replicate each method
bump.rep	Number of repetitions for bumping
parallel	Whether to run all reps in parallel
ncore	Number of cores to use
roundVal	How much to round cut points when calculating stability
stablelearner	Whether or not to use the stablelearner package to calculate stability
subset	Whether to subset
perc.sub	What fraction of data to put into train dataset. 1-frac.sub is allocated to test dataset. Defaults to 0.75
weights	Optional weights for each case.

Examples

```
## Not run:
library(MASS) # for boston data
data(Boston)
stab.out <- stable(formula=medv ~.,data=Boston,
  methods=c("rpart"),samp.method="cv",
  tuneLength=2, n.rep=5, parallel=TRUE)
stab.out

## End(Not run)
```

summary.dtree	<i>Summary results from dtree.</i>
---------------	------------------------------------

Description

Summary results from dtree.

Usage

```
## S3 method for class 'dtree'  
summary(object, ...)
```

Arguments

object	An object from dtree.
...	Other arguments.

Index

dtree, [2](#)

stable, [3](#)

summary.dtree, [4](#)