Package ‘forestError’

July 24, 2019

Type Package

Title Individualized Prediction Errors and Intervals for Random Forests

Version 0.0.0.9000

Author Benjamin Lu and Johanna Hardin

Maintainer Benjamin Lu <b.lu@berkeley.edu>

Description Estimates both conditional mean squared prediction errors and conditional prediction intervals for random forest predictions. The prediction errors and intervals are conditional in the sense that each error and interval is specific to the individual observation whose response is being predicted. More details can be found in Lu and Hardin (2019+) (in preparation).

Imports randomForest

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

R topics documented:

`quantForestError` *Quantify random forest prediction error*

Description

Estimates the conditional mean squared prediction errors and conditional prediction intervals of random forest predictions.

Usage

```
quantForestError(forest, X.train, X.test, alpha = 0.05,
  conservative = TRUE)
```

Arguments

forest	The random forest object being used for prediction.
X.train	A matrix or data.frame with the observations that were used to train forest; each row should be an observation, and each column should be a predictor variable.
X.test	A matrix or data.frame with the observations to be predicted; each row should be an observation, and each column should be a predictor variable.
alpha	The type-I error rate desired for the conditional prediction intervals; set to NA if no prediction intervals are desired. Defauls to 0.05.
conservative	A logical indicating whether a second set of conditional prediction intervals should be estimated in which ties in the empirical error distribution should be resolved conservatively or not. Defaults to TRUE.

Details

Three possible sets of outputs are possible from this function depending on the user's arguments for alpha and conservative.

The most minimal output is a list containing the random forest predictions of the test observations and the estimated conditional mean squared prediction errors associated with each. This can be obtained by setting alpha to NA.

The second possible output is a list that includes, in addition to the random forest predictions and the estimated prediction errors, conditional prediction intervals for each test observation with level specified by alpha. This can be obtained by setting alpha to a numeric value and setting conservative to FALSE.

The most extensive set of outputs includes, in addition to the above, a second set of prediction intervals generated by resolving ties in the empirical error distribution conservatively. Conservatively estimated prediction intervals may be desirable when the number of observations is relatively small. This output can be obtained by setting alpha to a numeric value and setting conservative to TRUE.

Value

A list with the following possible elements, each in the form of a vector, as described in the details section:

pred	The random forest predictions of the test observations
error	The estimated conditional mean square prediction errors of the random forest predictions
lower	The estimated lower bounds of the conditional prediction intervals for the test observations
upper	The estimated upper bounds of the conditional prediction intervals for the test observations
lowerCons	The conservatively estimated lower bounds of the conditional prediction intervals for the test observations
upperCons	The conservatively estimated upper bounds of the conditional prediction intervals for the test observations

Author(s)

Benjamin Lu <b.lu@berkeley.edu>; Johanna Hardin <jo.hardin@pomona.edu>

Examples

```
# load data
data(airquality)

# remove observations with missing predictor variable values
airquality <- airquality[complete.cases(airquality), ]

# get number of observations and the response column index
n <- nrow(airquality)
response.col <- 1

# split data into training and test sets
train.ind <- sample(1:n, n * 0.9, replace = FALSE)
Xtrain <- airquality[train.ind, -response.col]
Ytrain <- airquality[train.ind, response.col]
Xtest <- airquality[-train.ind, -response.col]
Ytest <- airquality[-train.ind, response.col]

# fit random forest to the training data
rf <- randomForest::randomForest(Xtrain, Ytrain, nodesize = 10, ntree = 500,
                                 keep.inbag = TRUE)

# get conditional mean squared prediction errors and prediction
# intervals for the test observations
test.preds <- quantForestError(rf, Xtrain, Xtest)

# get the same as above but without the conservative prediction
# intervals
test.preds <- quantForestError(rf, Xtrain, Xtest, conservative = FALSE)

# get just the mean squared prediction error estimates
test.preds <- quantForestError(rf, Xtrain, Xtest, alpha = NA)
```

Index