Package ‘forestError’

January 7, 2020

Type Package

Title A Unified Framework for Random Forest Prediction Error Estimation

Version 0.1.0

Author Benjamin Lu and Johanna Hardin

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

Description Estimates the conditional error distributions of random forest predictions and common parameters of those distributions, including conditional mean squared prediction errors, conditional biases, and conditional quantiles. This package is compatible with several existing packages that implement random forests in R.

Imports Rcpp,
foreach,
doParallel

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

LinkingTo Rcpp

R topics documented:

`perror` *Estimated conditional prediction error CDFs*

Description

Returns probabilities from the estimated conditional cumulative distribution function of the prediction error associated with each test observation.

Usage

```
perror(q, xs)
```

Arguments

q	A vector of quantiles.
xs	A vector of the indices of the test observations for which the conditional error CDFs are desired. Defaults to all test observations given in the call of quantForestError.

Value

If either q or xs has length one, then a vector is returned with the desired probabilities. If both have length greater than one, then a data.frame of probabilities is returned, with rows corresponding to the inputted xs and columns corresponding to the inputted q.

Author(s)

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

See Also

quantForestError

Examples

```
# get the probability that the error associated with each test
# prediction is less than -4 and the probability that the error
# associated with each test prediction is less than 7
perror(c(-4, 7))

# same as above but only for the first three test observations
perror(c(-4, 7), 1:3)
```

---

qerror   *Estimated conditional prediction error quantile functions*

Description

Returns quantiles of the estimated conditional error distribution associated with each test prediction.

Usage

```
qerror(p, xs)
```

Arguments

p	A vector of probabilities.
xs	A vector of the indices of the test observations for which the conditional error quantiles are desired. Defaults to all test observations given in the call of quantForestError.

Value

If either p or xs has length one, then a vector is returned with the desired quantiles. If both have length greater than one, then a data.frame of quantiles is returned, with rows corresponding to the inputted xs and columns corresponding to the inputted p.

Author(s)

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

See Also

quantForestError

Examples

```
# get the 0.25 and 0.8 quantiles of the error distribution
# associated with each test observation
qerror(c(0.25, 0.8))

# same as above but only for the first three test observations
qerror(c(0.25, 0.8), 1:3)
```

---

quantForestError *Quantify random forest prediction error*

Description

Estimates the conditional mean squared prediction errors, conditional biases, conditional prediction intervals, and conditional error distributions of random forest predictions.

Usage

```
quantForestError(forest, X.train, X.test, Y.train = NULL,
  what = c("mspe", "bias", "interval", "p.error", "q.error"),
  alpha = 0.05, n.cores = 1)
```

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.
Y.train	A vector of the responses of the observations that were used to train forest. Required if forest was created using ranger, but not if forest was created using randomForest, randomForestSRC, or quantregForest.

what — A vector of characters indicating what estimates are desired. Possible options are conditional mean squared prediction errors ("mspe"), conditional biases ("bias"), conditional prediction intervals ("interval"), conditional error distribution functions ("p.error"), and conditional quantile functions ("q.error").

alpha — The type-I error rate desired for the conditional prediction intervals; required if "interval" is included in what.

n.cores — Number of cores to use (for parallel computation).

Details

When training the random forest using randomForest, ranger, or quantregForest, keep.inbag must be set to TRUE. When training the random forest using randomForestSRC, membership must be set to TRUE.

The computation can be parallelized by setting the value of n.cores to be greater than 1.

The random forest predictions are always returned as a data.frame. Additional columns are included in the data.frame depending on the user's selections in the argument what. In particular, including "mspe" in what will add an additional column with the conditional mean squared prediction error of each test prediction to the data.frame; including "bias" in what will add an additional column with the conditional bias of each test prediction to the data.frame; and including "interval" in what will add to the data.frame two additional columns with the lower and upper bounds of a conditional prediction interval for each test prediction.

If "p.error" or "q.error" is included in what, then a list will be returned as output. The first element of the list, named "estimates", is the data.frame described in the above paragraph. The other one or two elements of the list are the estimated CDFs (perror) and/or the estimated quantile functions (qerror) of the conditional error distributions associated with the test predictions.

Value

A data.frame with one or more of the following columns, as described in the details section:

pred — The random forest predictions of the test observations

mspe — The estimated conditional mean square prediction errors of the random forest predictions

bias — The estimated conditional biases 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

In addition, one or both of the following functions, as described in the details section:

perror — The estimated cumulative distribution functions of the conditional error distributions associated with the test predictions

qerror — The estimated quantile functions of the conditional error distributions associated with the test predictions

Author(s)

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

See Also

perror, qerror

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(Xtrain, Ytrain, nodesize = 5,
                   ntree = 500, keep.inbag = TRUE)

# estimate conditional mean squared prediction errors,
# biases, prediction intervals, and error distribution
# functions for the test observations
test.errors <- quantForestError(rf, Xtrain, Xtest,
                                alpha = 0.05)

# do the same as above but in parallel
test.errors <- quantForestError(rf, Xtrain, Xtest,
                                alpha = 0.05,
                                n.cores = 4)

# estimate just the conditional mean squared prediction errors
# and prediction intervals for the test observations
test.errors <- quantForestError(rf, Xtrain, Xtest,
                                what = c("mspe", "interval"),
                                alpha = 0.05)

# estimate just the conditional error distribution
# functions for the test observations
test.errors <- quantForestError(rf, Xtrain, Xtest,
                                what = c("p.error", "q.error"))
```

Index