Package ‘interpretR’

March 19, 2016

Type Package

Title Binary Classifier and Regression Model Interpretation Functions

Version 0.2.4

Date 2016-03-19

Depends randomForest

Imports AUC, stats, graphics

Author Michel Ballings and Dirk Van den Poel

Maintainer Michel Ballings <michel.ballings@GMail.com>

Description Compute permutation- based performance measures and create partial dependence plots for (cross-validated) 'randomForest' and 'ada' models.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2016-03-19 22:55:06

R topics documented:

interpretR-package .................................................... 2
interpretRNews ............................................................ 2
parDepPlot ................................................................. 3
variableImportance ......................................................... 5

Index 7
Partial Dependence Plots and Permutation-Based Performance Measures

Description

Compute permutation-based performance measures (for binary classification) and create partial dependence plots (cross-validated classification and regression models). Currently only binary classification and regression models estimated with the package randomForest are supported. Binary classification models estimated with ada are also supported.

Details

Package: interpretR
Type: Package
Version: 0.2.1
Date: 2014-03-23
License: GPL (>= 2)

Author(s)

Authors: Michel Ballings and Dirk Van den Poel, Maintainer: <Michel.Ballings@GMail.com>

See Also

parDepPlot, variableImportance

Display the NEWS file

Description

interpretRNews shows the NEWS file of the interpretR package.

Usage

interpretRNews()

Author(s)

Authors: Michel Ballings and Dirk Van den Poel, Maintainer: <Michel.Ballings@GMail.com>
parDepPlot

See Also
parDepPlot

Examples

interpretRNews()

---

parDepPlot  Model interpretation functions: Partial Dependence Plots

Description

parDepPlot creates partial dependence plots for binary (cross-validated) classification models and regression models. Currently only binary classification models estimated with the packages randomForest and ada are supported. In addition randomForest regression models are supported.

Usage

parDepPlot(x.name, object, data, rm.outliers = TRUE, fact = 1.5, n.pt = 50, robust = FALSE, ci = FALSE, u.quant = 0.75, l.quant = 0.25, xlab = substr(x.name, 1, 50), ylab = NULL, main = if (any(class(object) %in% c("randomForest", "ada"))) paste("Partial Dependence on", substr(x.name, 1, 20)) else paste("Cross-Validated Partial Dependence on", substr(x.name, 1, 10)), logit = TRUE, ylim = NULL, ...)

Arguments

x.name the name of the predictor as a character string for which a partial dependence plot has to be created.

object can be a model or a list of cross-validated models. Currently only binary classification models built using the packages randomForest and ada are supported.

data a data frame containing the predictors for the model or a list of data frames for cross-validation with length equal to the number of models.

rm.outliers boolean, remove the outliers in x.name. Outliers are values that are smaller than max(Q1-fact*IQR,min) or greater than min(Q3+fact*IQR,max). Overridden if xlim is used.

fact factor to use in rm.outliers. The default is 1.5.

n.pt if x.name is a continuous predictor, the number of points that will be used to plot the curve.

robust if TRUE then the median is used to plot the central tendency (recommended when logit=FALSE). If FALSE the mean is used.

ci boolean. Should a confidence interval based on quantiles be plotted? This only works if robust=TRUE.
parDepPlot

u.quant Upper quantile for ci. This only works if ci=TRUE and robust=TRUE.
1.quant Lower quantile for ci. This only works if ci=TRUE and robust=TRUE.
xlab label for the x-axis. Is determined automatically if NULL.
ylab label for the y-axis.
main main title for the plot.
logit boolean. Should the y-axis be on a logit scale or not? If FALSE, it is recommended to set robust=TRUE. Only applicable for classification.
ylim The y limits of the plot
... other graphical parameters for plot.

Details
For classification, the response variable in the model is always assumed to take on the values \{0,1\}. Resulting partial dependence plots always refer to class 1. Whenever strange results are obtained the user has three options. First set rm.outliers=TRUE. Second, if that doesn't help, set robust=TRUE. Finally, if that doesn't help, the user can also try setting ci=TRUE. Areas with larger confidence intervals typically indicate problem areas. These options help the user tease out the root of strange results and converge to better parameter values.

Author(s)
Authors: Michel Ballings, and Dirk Van den Poel, Maintainer: <Michel.Ballings@GMail.com>

References
The code in this function uses part of the code from the partialPlot function in randomForest. It is expanded and generalized to support cross-validation and other packages.

See Also
variableImportance

Examples
library(randomForest)
#Prepare data
data(iris)
iris <- iris[1:100,]
iris$Species <- as.factor(ifelse(factor(iris$Species)=="setosa",0,1))

#Cross-validated models
#Estimate 10 models and create 10 test sets
data <- list()
rf <- list()
for (i in 1:10) {
  ind <- sample(nrow(iris),50)
  rf[[i]] <- randomForest(Species~, iris[ind,])
data[[i]] <- iris[-ind,]
}
variableImportance

parDepPlot(x.name="Petal.Width", object=rf, data=data)

# Single model
# Estimate a single model
ind <- sample(nrow(iris),50)
rf <- randomForest(Species~., iris[ind])
parDepPlot(x.name="Petal.Width", object=rf, data=iris[-ind])

variableImportance  Permutation- based Variable Importance Measures

Description

variableImportance produces permutation-based variable importance measures (currently only for binary classification models from the package randomForest and only for the performance measure AUROC)

Usage

variableImportance(object = NULL, xdata = NULL, ydata = NULL, CV = 3, measure = "AUROC", sort = TRUE)

Arguments

object A model. Currently only binary classification models from the package randomForest.
xdata A data frame containing the predictors for the model.
ydata A factor containing the response variable.
CV Cross-validation. How many times should the data be permuted and the decrease in performance be calculated? Afterwards the mean is taken. CV should be higher for very small samples to ensure stability.
measure Currently only Area Under the Receiver Operating Characteristic Curve (AUROC) is supported.
sort Logical. Should the results be sorted from high to low?

Details

Currently only binary classification models from randomForest are supported. Also, currently only AUROC is supported. Definition of MeanDecreaseAUROC: for the entire ensemble the AUROC is recorded on the provided xdata. The same is subsequently done after permuting each variable (iteratively, for each variable separately). Then the latter is subtracted from the former. This is called the Decrease in AUROC. If we do this for multiple CV, it becomes the Mean Decrease in AUROC.
Value

A data frame containing the variable names and the mean decrease in AUROC

Author(s)

Authors: Michel Ballings, and Dirk Van den Poel, Maintainer: <Michel.Ballings@GMail.com>

See Also

parDepPlot

Examples

# Prepare data
data(iris)
iris <- iris[1:100,]
iris$Species <- as.factor(ifelse(factor(iris$Species)=="setosa",0,1))

# Estimate model
library(randomForest)
ind <- sample(nrow(iris),50)
rf <- randomForest(Species~., iris[ind,])

# Obtain variable importances
variableImportance(object=rf, xdata=iris[-ind, names(iris) != "Species"], ydata=iris[-ind,]$Species)
Index

interpretR (interpretR-package), 2
interpretR-package, 2
interpretRNews, 2

parDepPlot, 2, 3, 3, 6

variableImportance, 2, 4, 5