Package 'randomForestExplainer'

July 11, 2020

Title Explaining and Visualizing Random Forests in Terms of Variable Importance

Version 0.10.1

Description A set of tools to help explain which variables are most important in a random forests. Various variable importance measures are calculated and visualized in different settings in order to get an idea on how their importance changes depending on our criteria (Hemant Ishwaran and Udaya B. Kogalur and Eiran Z. Gorodeski and Andy J. Minn and Michael S. Lauer (2010) <doi:10.1198/jasa.2009.tm08622>, Leo Breim

Depends R (>= 3.0)

License GPL

Encoding UTF-8

LazyData true

Imports data.table (>= 1.10.4), dplyr (>= 0.7.1), DT (>= 0.2), GGally (>= 1.3.0), ggplot2 (>= 2.2.1), ggrepel (>= 0.6.5), randomForest (>= 4.6.12), ranger(>= 0.9.0), reshape2 (>= 1.4.2), rmarkdown (>= 1.5)

Suggests knitr, MASS (>= 7.3.47), testthat

VignetteBuilder knitr

RoxygenNote 7.1.0

URL https://github.com/ModelOriented/randomForestExplainer

NeedsCompilation no

Author Aleksandra Paluszynska [aut], Przemyslaw Biecek [aut, ths], Yue Jiang [aut, cre] (<https://orcid.org/0000-0002-9798-5517>)

Maintainer Yue Jiang <rivehill@gmail.com>

Repository CRAN

Date/Publication 2020-07-11 20:30:02 UTC

R topics documented:

`explain_forest` *Explain a random forest*

Description

Explains a random forest in a html document using plots created by randomForestExplainer

Usage

```
explain_forest(
  forest,
  path = NULL,
  interactions = FALSE,
  data = NULL,
  vars = NULL,
  no_of_pred_plots = 3,
  pred_grid = 100,
  measures = NULL
)
```

Arguments

`forest`	A randomForest object created with the option localImp = TRUE
`path`	Path to write output html to
`interactions`	Logical value: should variable interactions be considered (this may be time-consuming)
`data`	The data frame on which forest was trained - necessary if interactions = TRUE
`vars`	A character vector with variables with respect to which interactions will be considered if NULL then they will be selected using the important_variables() function

no_of_pred_plots
: The number of most frequent interactions of numeric variables to plot predictions for

pred_grid
: The number of points on the grid of plot_predict_interaction (decrease in case memory problems)

measures
: A character vector specifying the importance measures to be used for plotting ggpairs

Value

A html document. If path is not specified, this document will be "Your_forest_explained.html" in your working directory

Examples

```
## Not run:
forest <- randomForest::randomForest(Species ~ ., data = iris, localImp = TRUE)
explain_forest(forest, interactions = TRUE)

## End(Not run)
```

---

important_variables *Extract k most important variables in a random forest*

Description

Get the names of k variables with highest sum of rankings based on the specified importance measures

Usage

```
important_variables(
  importance_frame,
  k = 15,
  measures = names(importance_frame)[2:min(5, ncol(importance_frame))],
  ties_action = "all"
)
```

Arguments

importance_frame
: A result of using the function measure_importance() to a random forest or a randomForest object

k
: The number of variables to extract

measures
: A character vector specifying the measures of importance to be used

ties_action
: One of three: c("none", "all", "draw"); specifies which variables to pick when ties occur. When set to "none" we may get less than k variables, when "all" we may get more and "draw" makes us get exactly k.

Value

A character vector with names of k variables with highest sum of rankings

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, localImp = TRUE, ntree = 300)
important_variables(measure_importance(forest), k = 2)
```

`measure_importance` *Importance of variables in a random forest*

Description

Get a data frame with various measures of importance of variables in a random forest

Usage

```
measure_importance(forest, mean_sample = "top_trees", measures = NULL)
```

Arguments

`forest`	A random forest produced by the function randomForest with option localImp = TRUE
`mean_sample`	The sample of trees on which mean minimal depth is calculated, possible values are "all_trees", "top_trees", "relevant_trees"
`measures`	A vector of names of importance measures to be calculated - if equal to NULL then all are calculated; if "p_value" is to be calculated then "no_of_nodes" will be too. Suitable measures for `classification` forests are: `mean_min_depth`, `accuracy_decrease`, `gini_decrease`, `no_of_nodes`, `times_a_root`. For `regression` forests choose from: `mean_min_depth`, `mse_increase`, `node_purity_increase`, `no_of_nodes`, `times_a_root`.

Value

A data frame with rows corresponding to variables and columns to various measures of importance of variables

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, localImp = TRUE, ntree = 300)
measure_importance(forest)
```

min_depth_distribution

Calculate minimal depth distribution of a random forest

Description

Get minimal depth values for all trees in a random forest

Usage

```
min_depth_distribution(forest)
```

Arguments

forest	A randomForest or ranger object

Value

A data frame with the value of minimal depth for every variable in every tree

Examples

```
min_depth_distribution(randomForest::randomForest(Species ~ ., data = iris, ntree = 100))
min_depth_distribution(ranger::ranger(Species ~ ., data = iris, num.trees = 100))
```

min_depth_interactions

Calculate mean conditional minimal depth

Description

Calculate mean conditional minimal depth with respect to a vector of variables

Usage

```
min_depth_interactions(
  forest,
  vars = important_variables(measure_importance(forest)),
  mean_sample = "top_trees",
  uncond_mean_sample = mean_sample
)
```

Arguments

forest	A randomForest object
vars	A character vector with variables with respect to which conditional minimal depth will be calculated; by default it is extracted by the important_variables function but this may be time consuming
mean_sample	The sample of trees on which conditional mean minimal depth is calculated, possible values are "all_trees", "top_trees", "relevant_trees"
uncond_mean_sample	The sample of trees on which unconditional mean minimal depth is calculated, possible values are "all_trees", "top_trees", "relevant_trees"

Value

A data frame with each observation giving the means of conditional minimal depth and the size of sample for a given interaction

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, ntree = 100)
min_depth_interactions(forest, c("Petal.Width", "Petal.Length"))
```

---

plot_importance_ggpairs

Plot importance measures with ggpairs

Description

Plot selected measures of importance of variables in a forest using ggpairs

Usage

```
plot_importance_ggpairs(
  importance_frame,
  measures = NULL,
  main = "Relations between measures of importance"
)
```

Arguments

importance_frame	A result of using the function measure_importance() to a random forest or a randomForest object
measures	A character vector specifying the measures of importance to be used
main	A string to be used as title of the plot

Value

A ggplot object

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, localImp = TRUE, ntree = 200)
frame <- measure_importance(forest, measures = c("mean_min_depth", "times_a_root"))
plot_importance_ggpairs(frame, measures = c("mean_min_depth", "times_a_root"))
```

---

plot_importance_rankings

Plot importance measures rankings with ggpairs

Description

Plot against each other rankings of variables according to various measures of importance

Usage

```
plot_importance_rankings(
  importance_frame,
  measures = NULL,
  main = "Relations between rankings according to different measures"
)
```

Arguments

importance_frame	A result of using the function measure_importance() to a random forest or a randomForest object
measures	A character vector specifying the measures of importance to be used.
main	A string to be used as title of the plot

Value

A ggplot object

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, localImp = TRUE, ntree = 300)
frame <- measure_importance(forest, measures = c("mean_min_depth", "times_a_root"))
plot_importance_ggpairs(frame, measures = c("mean_min_depth", "times_a_root"))
```

`plot_min_depth_distribution` *Plot the distribution of minimal depth in a random forest*

Description

Plot the distribution of minimal depth in a random forest

Usage

```
plot_min_depth_distribution(
  min_depth_frame,
  k = 10,
  min_no_of_trees = 0,
  mean_sample = "top_trees",
  mean_scale = FALSE,
  mean_round = 2,
  main = "Distribution of minimal depth and its mean"
)
```

Arguments

`min_depth_frame`	A data frame output of min_depth_distribution function or a randomForest object
`k`	The maximal number of variables with lowest mean minimal depth to be used for plotting
`min_no_of_trees`	The minimal number of trees in which a variable has to be used for splitting to be used for plotting
`mean_sample`	The sample of trees on which mean minimal depth is calculated, possible values are "all_trees", "top_trees", "relevant_trees"
`mean_scale`	Logical: should the values of mean minimal depth be rescaled to the interval [0,1]?
`mean_round`	The number of digits used for displaying mean minimal depth
`main`	A string to be used as title of the plot

Value

A ggplot object

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, ntree = 300)
plot_min_depth_distribution(min_depth_distribution(forest))
```

plot_min_depth_interactions

Plot the top mean conditional minimal depth

Description

Plot the top mean conditional minimal depth

Usage

```
plot_min_depth_interactions(
  interactions_frame,
  k = 30,
 main = paste0("Mean minimal depth for ", paste0(k, " most frequent interactions"))
)
```

Arguments

interactions_frame	A data frame produced by the min_depth_interactions() function or a randomForest object
k	The number of best interactions to plot, if set to NULL then all plotted
main	A string to be used as title of the plot

Value

A ggplot2 object

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, ntree = 100)
plot_min_depth_interactions(min_depth_interactions(forest, c("Petal.Width", "Petal.Length")))
```

plot_multi_way_importance

Multi-way importance plot

Description

Plot two or three measures of importance of variables in a random fores. Choose importance measures from the colnames(importance_frame).

Usage

```
plot_multi_way_importance(
  importance_frame,
  x_measure = "mean_min_depth",
  y_measure = "times_a_root",
  size_measure = NULL,
  min_no_of_trees = 0,
  no_of_labels = 10,
  main = "Multi-way importance plot"
)
```

Arguments

importance_frame	A result of using the function measure_importance() to a random forest or a randomForest object
x_measure	The measure of importance to be shown on the X axis
y_measure	The measure of importance to be shown on the Y axis
size_measure	The measure of importance to be shown as size of points (optional)
min_no_of_trees	The minimal number of trees in which a variable has to be used for splitting to be used for plotting
no_of_labels	The approximate number of best variables (according to all measures plotted) to be labeled (more will be labeled in case of ties)
main	A string to be used as title of the plot

Value

A ggplot object

Examples

```
forest <- randomForest::randomForest(Species ~ ., data = iris, localImp = TRUE)
plot_multi_way_importance(measure_importance(forest))
```

---

plot_predict_interaction

Plot the prediction of the forest for a grid of values of two numerical variables

---

Description

Plot the prediction of the forest for a grid of values of two numerical variables

Usage

```
plot_predict_interaction(
  forest,
  data,
  variable1,
  variable2,
  grid = 100,
 main = paste0("Prediction of the forest for different values of ", paste0(variable1,
    paste0(" and ", variable2))),
  time = NULL
)
```

Arguments

forest	A randomForest or ranger object
data	The data frame on which forest was trained
variable1	A character string with the name a numerical predictor that will on X-axis
variable2	A character string with the name a numerical predictor that will on Y-axis
grid	The number of points on the one-dimensional grid on x and y-axis
main	A string to be used as title of the plot
time	A numeric value specifying the time at which to predict survival probability, only applies to survival forests. If not specified, the time closest to predicted median survival time is used

Value

A ggplot2 object

Examples

```
forest <- randomForest::randomForest(Species ~., data = iris)
plot_predict_interaction(forest, iris, "Petal.Width", "Sepal.Width")
forest_ranger <- ranger::ranger(Species ~., data = iris)
plot_predict_interaction(forest, iris, "Petal.Width", "Sepal.Width")
```

Index