Package ‘xgrove’

September 13, 2023

Title Explanation Groves
Version 0.1-7
Date 2023-09-12
Author Gero Szepannek [aut, cre]
Maintainer Gero Szepannek <gero.szepannek@web.de>
License GPL (>= 2)
Encoding UTF-8
Imports gbm, dplyr, rpart, rpart.plot
Suggests pdp, randomForest
RoxygenNote 7.2.0
NeedsCompilation no
Repository CRAN
Date/Publication 2023-09-13 09:40:02 UTC

R topics documented:

plot.sgtree ......................................................... 2
plot.xgrove .......................................................... 3
sgtree ................................................................. 4
upsilon ................................................................. 5
xgrove ................................................................. 6

Index 9
**plot.sgtree**

**Plot surrogate tree statistics**

**Description**

Plot statistics of surrogate trees to analyze complexity vs. explanatory power.

**Usage**

```r
## S3 method for class 'sgtree'
plot(x, abs = "rules", ord = "upsilon", ...)
```

**Arguments**

- `x`: An object of class `sgtree`.
- `abs`: Name of the measure to be plotted on the x-axis, either "trees", "rules", "upsilon" or "cor".
- `ord`: Name of the measure to be plotted on the y-axis, either "trees", "rules", "upsilon" or "cor".
- `...`: Further arguments passed to `plot`.

**Value**

No return value.

**Author(s)**

<gero.szepannek@web.de>

**Examples**

```r
library(randomForest)
library(pdp)
data(boston)
set.seed(42)
rf <- randomForest(cmedv ~ ., data = boston)
data <- boston[, -3] # remove target variable
ntrees <- c(4, 8, 16, 32, 64, 128)
xg <- xgrove(rf, data, ntrees)
xg
plot(xg)
```
**plot.xgrove**

*Plot surrogate grove statistics*

**Description**

Plot statistics of surrogate groves to analyze complexity vs. explanatory power.

**Usage**

```r
## S3 method for class 'xgrove'
plot(x, abs = "rules", ord = "upsilon", ...)
```

**Arguments**

- `x` An object of class `xgrove`.
- `abs` Name of the measure to be plotted on the x-axis, either "trees", "rules", "upsilon" or "cor".
- `ord` Name of the measure to be plotted on the y-axis, either "trees", "rules", "upsilon" or "cor".
- `...` Further arguments passed to `plot`.

**Value**

No return value.

**Author(s)**

<gero.szepannek@web.de>

**Examples**

```r
library(randomForest)
library(pdp)
data(boston)
set.seed(42)
rf <- randomForest(cmedv ~ ., data = boston)
data <- boston[, -3] # remove target variable
ntrees <- c(4, 8, 16, 32, 64, 128)
xg <- xgrove(rf, data, ntrees)
xg
plot(xg)
```
**Description**

Compute surrogate trees of different depth to explain predictive machine learning model and analyze complexity vs. explanatory power.

**Usage**

```
sgtree(model, data, maxdeps = 1:8, cparam = 0, pfun = NULL, ...)
```

**Arguments**

- **model**: A model with corresponding predict function that returns numeric values.
- **data**: Data that must not (!) contain the target variable.
- **maxdeps**: Sequence of integers: Maximum depth of the trees.
- **cparam**: Complexity parameter for growing the trees.
- **pfun**: Optional predict function `function(model, data) returning a real number. Default is the predict() method of the model.
- **...**: Further arguments to be passed to `rpart.control` or the `predict()` method of the model.

**Details**

A surrogate grove is trained via gradient boosting using `rpart` on `data` with the predictions of using the `model` as target variable. Note that `data` must not contain the original target variable!

**Value**

List of the results:

- **explanation**: Matrix containing tree sizes, rules, explainability $\Upsilon$ and the correlation between the predictions of the explanation and the true model.
- **rules**: List of rules for each tree.
- **model**: List of the `rpart` models.

**Author(s)**

<gero.szepannek@web.de>

**References**

- Szepannek, G. and Laabs, B.H. (2023): Can’t see the forest for the trees – analyzing groves to explain random forests, Behaviormetrika, submitted.
Examples

```r
library(randomForest)
library(pdp)
data(boston)
set.seed(42)
rf <- randomForest(cmedv ~ ., data = boston)
data <- boston[, -3] # remove target variable
maxds <- 1:7
st <- sgtree(rf, data, maxds)
st # rules for tree of depth 3
st$rules[["3"]]
# plot tree of depth 3
rpart.plot::rpart.plot(st$model[["3"]])
```

### `upsilon`

<table>
<thead>
<tr>
<th>Explainability</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>upsilon</code></td>
</tr>
</tbody>
</table>

#### Description

Compute explainability given predicted data of the model and an explainer.

#### Usage

```r
upsilon(porig, pexp)
```

#### Arguments

- `porig` An object of class `xgrove`.
- `pexp` Name of the measure to be plotted on the x-axis, either "trees", "rules", "upsilon" or "cor".

#### Value

Numeric explainability `upsilon`.

#### Author(s)

<gero.szepannek@web.de>

#### References

Examples

```r
library(randomForest)
library(pdp)
data(boston)
set.seed(42)
# Compute original model
rf <- randomForest(cmedv ~ ., data = boston)
data <- boston[, -3] # remove target variable
# Compute predictions
porig <- predict(rf, data)

# Compute surrogate grove
xg <- xgrove(rf, data)
pexp <- predict(xg$model, data, n.trees = 16)
upsilon(porig, pexp)
```

---

**xgrove**

*Explanation groves*

**Description**

Compute surrogate groves to explain predictive machine learning model and analyze complexity vs. explanatory power.

**Usage**

```r
xgrove(
  model,
  data,
  ntrees = c(4, 8, 16, 32, 64, 128),
  pfun = NULL,
  shrink = 1,
  b.frac = 1,
  seed = 42,
  ...
)
```

**Arguments**

- **model**: A model with corresponding predict function that returns numeric values.
- **data**: Data that must not (!) contain the target variable.
- **ntrees**: Sequence of integers: number of boosting trees for rule extraction.
- **pfun**: Optional predict function function(model, data) returning a real number. Default is the predict() method of the model.
xgrove

shrink
Sets the shrinkage argument for the internal call of \textit{gbm}. As the model usually has a deterministic response the default is 1 different to the default of \textit{gbm} applied train a model based on data.

b.frac
Sets the bag.fraction argument for the internal call of \textit{gbm}. As the model usually has a deterministic response the default is 1 different to the default of \textit{gbm} applied train a model based on data.

seed
Seed for the random number generator to ensure reproducible results (e.g. for the default bag.fraction < 1 in boosting).

... Further arguments to be passed to \textit{gbm} or the \texttt{predict()} method of the model.

Details
A surrogate grove is trained via gradient boosting using \textit{gbm} on data with the predictions of using of the model as target variable. Note that data must not contain the original target variable! The boosting model is trained using stumps of depth 1. The resulting interpretation is extracted from \textit{pretty.gbm.tree}.

Value
List of the results:

- \textit{explanation} Matrix containing tree sizes, rules, explainability \( \Upsilon \) and the correlation between the predictions of the explanation and the true model.
- \textit{rules} Summary of the explanation grove: Rules with identical splits are aggregated. For numeric variables any splits are merge if they lead to identical parititions of the training data
- \textit{groves} Rules of the explanation grove.
- \textit{model} \textit{gbm} model.

Author(s)
<gero.szepannek@web.de>

References

Examples
library(randomForest)
library(pdp)
data(boston)
set.seed(42)
rf <- randomForest(cmedv~., data = boston)
data <- boston[, -3] # remove target variable
ntrees <- c(4, 8, 16, 32, 64, 128)
xg <- xgrove(rf, data, ntrees)
xg
plot(xg)
Index

gbm, 7
plot.sgtree, 2
plot.xgrove, 3
pretty.gbm.tree, 7
rpart, 4
rpart.control, 4
sgtree, 4
upsilon, 5
xgrove, 6