

EIX :: Explain Interactions in XGBoost

MI

The **EIX** (**Explain Interactions in XGBoost**) package supports structure mining from XGBoost and LightGBM models.

Key functionalities of this package cover: visualisation of tree-based ensembles models, identification of interactions, measuring of variable importance, measuring of interaction importance, explanation of single prediction with break down plots (based on **xgboostExplainer** and **breakDown** packages).

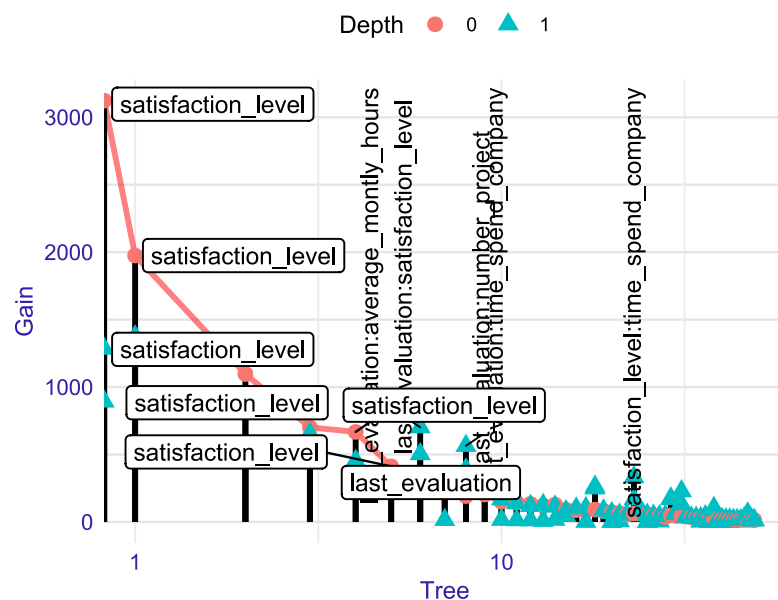
```
devtools::install_github("ekarbowiak/EIX")
library("EIX")
```

```
library("Matrix")
sm <- sparse.model.matrix(left ~ . - 1, data = HR_data)
library("xgboost")
param <- list(objective = "binary:logistic", max_depth = 2)
xgb_model <- xgboost(sm, params = param, label =
HR_data[, left] == 1, nrounds = 50)
```

Visualisation of model

The **lollipop plot** shows the model with labels of the most important variables and interactions.

```
loli <- lollipop(xgb_model, sm)
plot(loli, labels = "topAll", log_scale = TRUE)
```



Importance of variables and interactions

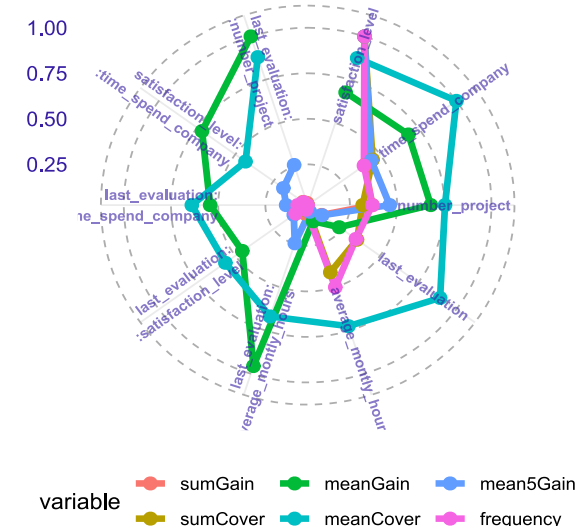
The **importance table** includes different measures of importance for variables and interactions. It is possible to visualise it in two ways using **plot()** function.

- option = "variables"** - object consists only single variables
- option = "interactions"** - only interactions
- option = "both"** - object shows importance both single variables and interactions. Default option.

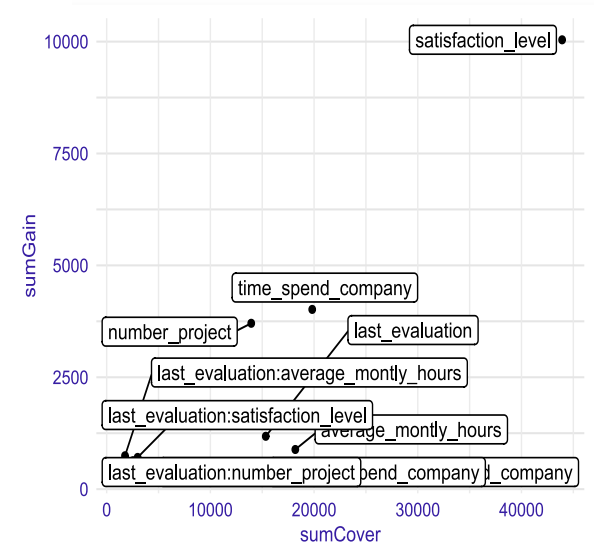
Available measures: sumGain, sumCover, mean5Gain (mean gain from 5 occurrences of given variable with the highest gain), meanGain, meanCover, frequency

For single variables additionally: numberOfRoots, meanDepth, weightedRoot (last two weighted by Gain of variable)

```
imp <- importance(xgb_model, sm,
option = "both")
plot(imp, top = 10)
```



```
plot(imp, radar = FALSE,
xmeasure = "sumCover", ymeasure =
"sumGain", top = 10)
```

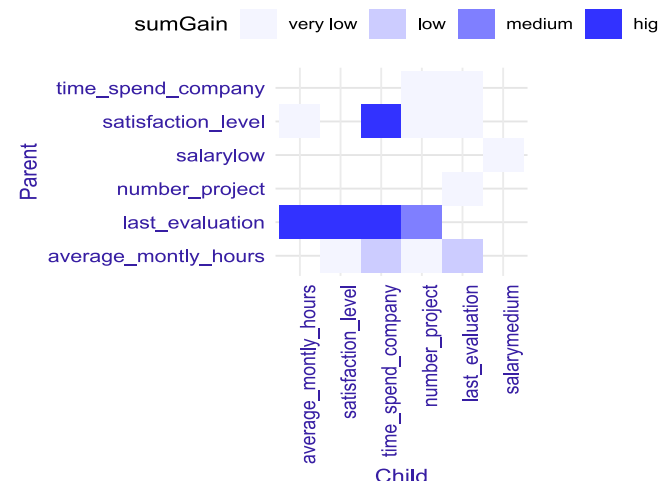


Interactions

The **interactions table** makes the ranking of interactions in the model. Importance of interactions it can also be visualised with **plot()** function.

```
inter <- interactions(xgb_model, sm,
option = "interactions")
plot(inter)
```

- option = "interactions"** - these pairs of variables in which variable on the bottom (child) has higher gain than variable on the top (parent). Default option.
- option = "pairs"** - all pairs of variables, which occur in the model one above the other.



Explanation of a single prediction

The **waterfall plot** shows, which variables have influence on the prediction of a single observation.

```
require("iBreakDown")
new_observation <- sm[9,]
data <- HR_data[9,-7]
wf <- waterfall(xgb_model, new_observation, data, option = "interactions")
plot(wf)
```

