IntTreeR

Interactive Classification Tree Package for R

https://github.com/IntTreeR/IntTreeR

AGENDA

- About InTreeR

- Who developed IntTreeR

- Function overview

- Install InTreeR

- Build your first interactive tree

- Contact us

InTreeR- Interactive Tree for R

About IntTreeR

- Represents an open source R package to built classification trees

- Built your own decision trees in an interactive mode like you may know from commercial tools like IBM SPSS Modeler®

- The implemented algorithm follows CART Algorithm by Leo Breiman (1984) et. Al.

- The printing of the trees is realized with the "Partykit - Toolkit for recursive partytioning"

- Additionally it includes the powerful opportunity to impute Missing Values via Surrogate Splits

Who developed IntTreeR?

- IntTreeR stands for Interactive Tree (R)

- Furthermore the package name is an allusion to the hometown (Trier) of the development Team

- It was developed by four masters students at Trier – University of applied sciences in Germany

© Presseamt Trier

Function overview

- The whole package includes seven functions*
  - initTree(*data, v.target, b.automated, theta*)
  - splitTree(*tree.obj, split.point, split.feature, max.surrogates, custom.split*)
  - plotTree(*tree.obj*)
  - printTree(*tree.obj*)
  - splitOptions(*tree.obj, split.point*)
  - cutTree(*tree.obj, cut.point*)
  - doPrediction(*tree.obj, predict.data*)

* For details please take a look at our documentation in R

Install IntTreeR

- If you want to install the package you can use these instructions:

```
install.packages('devtools')
y
devtools::install_github('IntTreeR/IntTreeR')
library(IntTreeR)
```

- … or visit us on Github: https://github.com/IntTreeR/IntTreeR

Build your first interactive tree (1)

- The first step is to load your dataset into a data.frame

- In the current implementation it is only possible to built a tree with a binary target variable (e.g. good/bad, true/false, 0/1 …)

```
dat = read.csv("/Users/FlorianMatthies/Desktop/kredit_bereinigt.csv",sep=";",dec=",")
head(dat)
```

rueckzah	kreditbe	geschlec	alter	kinderza	anschrif	buerge	arbeitsd	einkomme	verfeink	miete	pkw	ausgaben
1	24000	1	27	1	1	1	2	2900	1335	330	1	1565
1	15000	0	28	0	1	0	20	2000	1100	150	0	900
1	14500	1	68	0	1	1	6	2415	865	750	0	1550
1	30000	1	22	0	0	1	6	2600	920	530	1	1680
1	4000	0	40	1	1	0	20	2000	350	0	0	1650
1	30000	1	42	2	1	1	4	3580	1592	515	1	2145

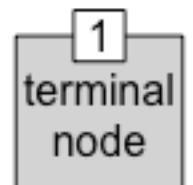# Build your first interactive tree (2)

- After you have loaded your dataset into R you can already create your Root element of the tree.

- The root element is built with the *initTree()* function

```
meinBaum = initTree(data= dat,v.target= "rueckzah")
```

- Now you can take a first look at your "tree" by calling the *plotTree()* function

```
plotTree(tree.obj= meinBaum)
```

Your tree ☺

```
1
terminal
node
```

Trier University
of Applied Sciences

H   O C H
S C   H U L E
T R IE R

Build your first interactive tree (3)

- Now you need to decide which attribute you want to use for your very first split. For this reason you can call the *splitOptions* function

- Conventional Decision Tree implementations will choose automatically the attribute with the highest improvement value for their splits. But our package is an interactive implementation, so YOU choose the split-attribute.

```
splitOptions(tree.obj = meinBaum,split.point= 1)
```

|	Attribute	Improvement
3	alter	0.173591485
7	arbeitsd	0.142011394
5	anschrif	0.095228160
9	verfeink	0.090776705
1	kreditbe	0.082450774
8	einkomme	0.065130360
10	miete	0.057165168
11	pkw	0.048952287
12	ausgaben	0.047982408
4	kinderza	0.015610714

Build your first interactive tree (4)

- Now you can expand your tree by using the *splitTree* function (first split always at split.point 1)*

```
meinBaum = splitTree(tree.obj= meinBaum, split.point= 1,split.feature= "alter")
```

- … and again you can take a look at your current tree with *plotTree(tree.obj)*

```
plotTree(tree.obj= meinBaum)
```

* With the optional parameter custom.split it is possible to use an own point of splitting by your own (e.g. custom.split = 20 will result in splitting at the age of 20 instead of the calculated splitting point 39.
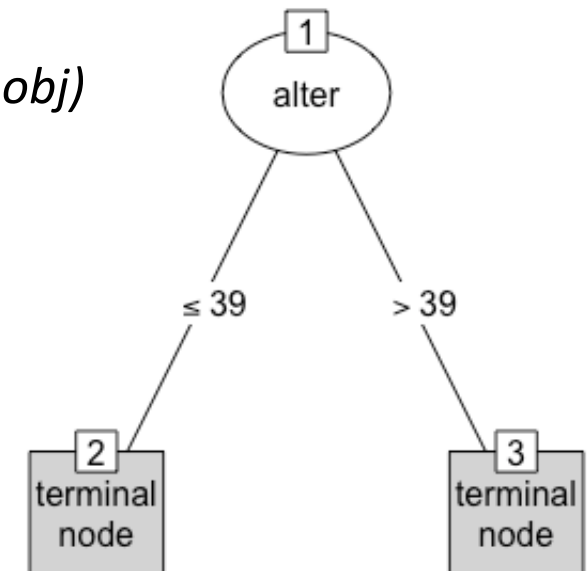
Build your first interactive tree (5)

- At this point your tree only consists of three nodes

- Therefore you have to decide at which point your next split sho at node 2

```
splitOptions(meinBaum,2)
meinBaum = splitTree(meinBaum,2, "miete",max.surrogates= 3)
plotTree(meinBaum)
```
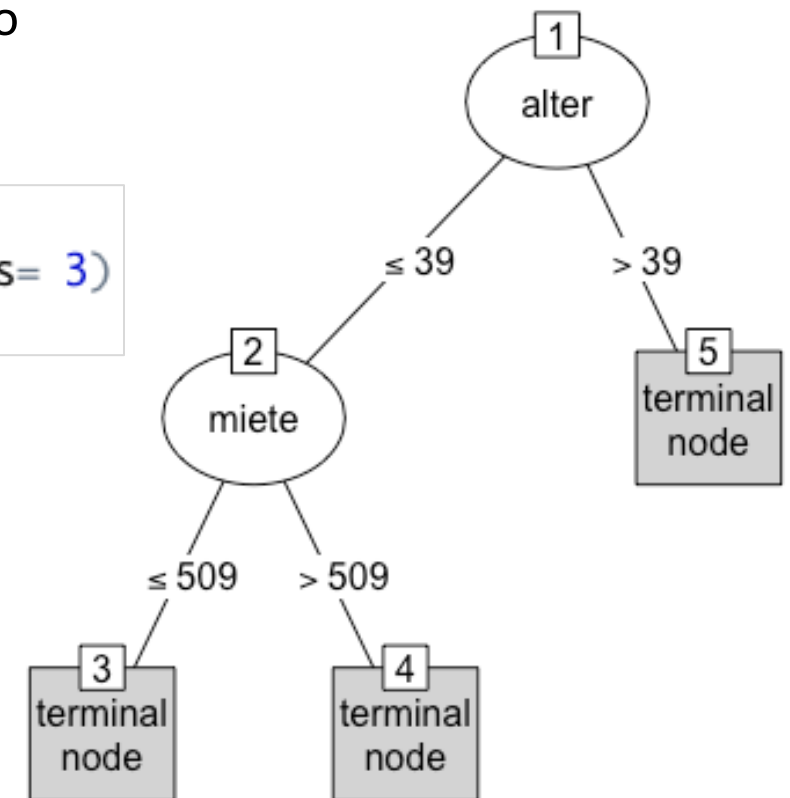
Build your first interactive tree (6)

- If you want to **customize your tree**, you could use all functionalities of the Partykit package.

- For example you could write your own panel functions to show diagrams / statistics in the node

InTreeR- Interactive Tree for R
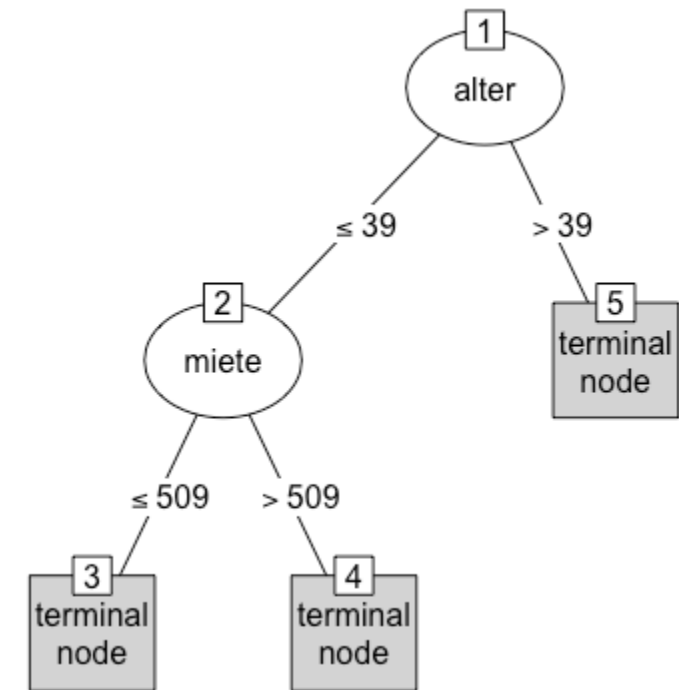
Other examples

- Which data is in what node?

```
meinBaum[[3]]$data
```

- Cut the tree*

```
meinBaum = cutTree(tree.obj = meinBaum,cut.point = 2)
```

- Build a non interactive tree

```
meinBaum = initTree(dat, "rueckzah", auto = TRUE)
```

*please note that it is only possible to cut at the lowest level.
Therefore you maybe need to cut multiple times.*

Contact us

- If have any experience with our package (positive or negative) or you would ask/tell us everything else, we would appreciate it if you share it with us ☺

- You can contact us via inttreer@gmx.de