

Lab12. Decision Trees and Forests.

Variable importance

We will use the library `party`. However, there is a number of other packages for classification and regression tree-based approach (CART): `randomForest`, `rpart`, `crat`, `maptree`, `partykit` and other.

```
library(party)
```

1. Consonant drop in Russian

Our student Varvara Sveshnikova wrote her BA paper on two cases of the consonant drop:

(a) when in the complex `-stvov-` (like in *beschinstVovat* 'to riot') another labial consonant is pronounced after it, and

(b) when no consonant follows (in two contexts: *beschinstVuju* 'I riot', *beschinstVo_* 'roistering').

The dataset

(https://raw.githubusercontent.com/agricolamz/r_on_line_course_data/master/Sveshnikova.2016.v.elision.csv) includes the following data:

`v.elision` — elision of [v] / no elision;

`group` — a group of test words, first (*beschinstvovat*), second (*beschinstvuju*), third (*beschinstvo*);

`word` — root under analysis;

`position` — phrase position: strong, under logical stress (*I am not CRYING, I resent*), weak (*He ALWAYS likes to cry*).

Fit a CART model, using `ctree()` function, predicting `v.elision` variable by all others.

1.1 Visualize a model using `plot()` function. What is the number of observation in node 6?

1.2 Visualize a model using `print()` function. Which split have a statistic 14.01?

1.3 Predict a value of `v.elision` for word with a root "noneч" in a third group, in a strong position.

Fit a `cforest` model using additional argument `controls=cforest_unbiased(ntree=1000, mtry=3)`.

1.4 Predict a value of `v.elision` for word with a root "noneч" in a third group, in a strong position using `cforest` model.

You need to add an argument `OOB=TRUE`, e. g. `yes`

1.5 Calculate a variable importance for a group variable in the random forest model using `varimp()` function.

Code to use:

```
df <- read.csv("https://raw.githubusercontent.com/agricolamz/r_on_line_course_data/master/Sveshnikova.2016.v.elision.csv")
fit <- party::ctree(v.elision~., data = df) # use the argument controls = ctree_control(...) to control the max depth etc.
plot(fit)
```

