Homework Assignment 1

Data Science and Machine Learning 2 - CEU 2018

Due date: 2018-03-19 8:00 a.m.

General information

The due date of this homework assignment is 2018-03-19 8:00 a.m. You are required to submit two files to Moodle: an `.Rmd` file and the rendered `.html` file with your solutions. Please also include code to the `.html` as well (use the default settings).

Please give short (2-3 sentences) interpretations, explanations to your answers, not only the program code and outputs.

Grading: 13 points can be earned in total. However, 10 points count as 100%. You can earn more than 100%, that is, giving a perfect answer to all questions means 130%.

```
library(data.table)
library(caret)
library(rpart)
library(rpart.plot)
library(xgboost)
library(randomForest)
library(gbm)
library(ISLR)
library(skimr)
library(ROCR)
```

1. Classification tree model (3 points)

In this problem you are going to work with the `OJ` dataset from the `ISLR` package. This dataset records purchases of two types of orange juices and presents customer and product characteristics as features. The goal is to predict which of the juices is chosen in a given purchase situation. See `?ISLR::OJ` for a description of the variables.

```
data <- data.table(OJ)
```

a) Create a training data of 75% and keep 25% of the data as a test set.

b) Build a classification tree, determining the optimal complexity parameter via 10-fold cross validation.

- Use values for the complexity parameter ranging between 0.001 and 0.1.
- the selection criterion should be based on AUC
- Use the "one standard error" rule to select the final model

c) Plot the final model and interpret the result. How would you predict a new observation?

d) Evaluate the final model on the test set. Is the AUC close to what we got via cross-validation?

2. Tree ensemble models (6 points)

For the same problem analyzed in Problem 1, investigate tree ensemble models:

- random forest
- gradient boosting machine
- XGBoost

a) Try various tuning parameter combinations and select the best model using cross-validation. (This time when doing hyperparameter tuning, simply choose the best model instead of applying the oneSE rule.)
b) Compare different models with the `resamples` function (make sure to set the same seed before model training for all 3 models). Is any of these giving significantly different predictive power than the others?
c) Choose the best model and plot ROC curve for the best model on the test set. Calculate and interpret AUC.
d) Inspect variable importance plots for the 3 models. Are similar variables found to be the most important for the 3 models?

3. Variable importance profiles (4 points)

Use the `Hitters` dataset and predict `log_salary` just like we did it in class.

```
data <- data.table(Hitters)
data <- data[!is.na(Salary)]
data[, log_salary := log(Salary)]
data[, Salary := NULL]
```

a) train two random forest models: one with `mtry = 2` and another with `mtry = 10` (use the whole dataset and don't use cross-validation). Inspect variable importance profiles. What do you see in terms of how important the first few variables are relative to each other?
b) One of them is more extreme in terms of how the most important and the next ones relate to each other. Give an intuitive explanation how `mtry` relates to relative importance of variables in random forest models.
c) In the same vein, estimate two `gbm` models and set `bag.fraction` to 0.1 first and to 0.9 in the second. The `tuneGrid` should consist of the same values for the two models (a dataframe with one row):

- n.trees = 500
- interaction.depth = 5
- shrinkage = 0.1
- n.minobsinnode = 5

Compare variable importance plots for the two models. What is the meaning of `bag.fraction`? Based on this, why is one variable importance profile more extreme than the other?