Fachbereich AIHI
Universität Salzburg
Univ.-Prof. Dr. Wolfgang Trutschnig

Salzburg, December 2025
WS 25/26

Exercise Sheet 05 zu 'Wissenschaftliches Rechnen - R'

We are working with decision trees, first consider two exercises on decision trees and then study, how trees/random forests can also be used for timeseries forecasting.

Exercise 22.
Consider the following small dataset that tracks a person's TV watching preferences. The columns `comedy`, `doctors`, `lawyers` and `guns` are the predictor variables, indicating if the TV show featured comedy, doctors, lawyers and/or guns. The column `likes` (target variable) specifies, if the person liked the TV show.

comedy	doctors	lawyers	guns	likes
FALSE	TRUE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	FALSE	FALSE
FALSE	TRUE	FALSE	FALSE	FALSE
FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	TRUE	TRUE
TRUE	TRUE	FALSE	FALSE	TRUE
FALSE	TRUE	FALSE	TRUE	FALSE
TRUE	TRUE	FALSE	TRUE	TRUE
TRUE	TRUE	FALSE	TRUE	TRUE
TRUE	TRUE	FALSE	TRUE	TRUE

Answer the following simple questions:

1. Compute the entropy for the `likes` variable, $H(\texttt{likes})$
2. Compute the conditional entropy $H(\texttt{likes}|\texttt{doctors} = 1)$
3. Compute the information gain $I(\texttt{likes}, \texttt{doctors} = 1)$
4. Compute the expected information gain $I(\texttt{likes}, \texttt{doctors})$
5. What is the optimal depth-0 decision tree (having only one leaf node and no tests)?
6. What is the optimal depth-1 decision tree (having one inner node and two leaves)?

Exercise 23.
We consider the *indonesian nutrition dataset* (downloadable on Blackboard to avoid copyright issues[i]) and want to check the performance of a simple regression tree by completing the following steps:

1. Randomly split the data into a training and a test dataset (70:30).
2. Fit a regression tree (calories $\sim$ fat + proteins + carbohydrate) to the training data, predict the calories for the test data, and calculate the residuals. Work with `rpart` and set `method="anova"`.

[i] please login to blackboard and download the file there

3. Illustrate the obtained results by boxplots (for the residuals) in ggplot2 and include the mean absolute errors (MAE) in the boxplots.

4. Repeat the steps 1.-4. $R = 1.000$ times and produce another boxplot summarizing the MAEs for the tree models.

Exercise 24.
We work with the ATM dataset already discussed in `07_Sprout` (Slides page 43-46, R-Snippet, lines 120+), proceed as sketched on page 44 in the slides, and only use the following features as predictors: `weekday, day.of.month, month, festivity`[ii], `pre-festivity`[iii]

1. Use the data from 2008-01-01 till 2009-01-11 to train a model for predicting $d = 1$ day ahead.

2. Predict the withdrawn amount on 2009-01-12.

3. Use the data from 2008-01-01 till 2009-01-12 to train a model.

4. Predict the withdrawn amount on 2009-01-13.

5. ...proceed analogously till forecasting the value for 2009-12-31.

6. Calculate the MAE (mean absolute error) of all 353 forecasts.

Exercise 25.
Use the same predictors as in the previous exercise but this time forecast $d = 7$ days ahead:

1. Use the data from 2008-01-01 till 2009-01-11 to train a model for predicting $d = 7$ day ahead.

2. Predict the withdrawn amount on 2009-01-18.

3. Use the data from 2008-01-01 till 2009-01-12 to train a model.

4. Predict the withdrawn amount on 2009-01-19.

5. ...proceed analogously forecasting the value for 2009-12-31.

6. Calculate the MAE (mean absolute error) of all 346 forecasts.

Exercise 26.
Notice that in the previous two exercises we completely ignored the withdrawn amounts of the past couple of days. Obviously the forecasts should improve if we include this information. Build a model (tree or Random Forest) using the following extended set of features:

[ii] yes/no; 1 means 'yes', all other values 'no'
[iii] yes/no; 1.5 means 'yes', all other values 'no'

`weekday, day.of.month, month, festivity, pre-festivity, sum.out.m0, sum.out.m1, sum.out.m2, sum.out.m3`[iv], `sum.out.m4, sum.out.m5,sum.out.m6,sum.out.m7`
whereby, e.g., the feature `sum.out.m6` is sum_out 6 days ago.

1. Use the data from 2008-01-01 till 2009-01-11 to train the model for predicting $d = 1$ day ahead. Notice that in this case `sum.out.m0` is the withdrawn amount on day 2009-01-11, `sum.out.m1` is the withdrawn amount on day 2009-01-10, `sum.out.m2` is the withdrawn amount on day 2009-01-09, and so on. Use the learned dplyr tools to do the required data wrangling.
2. Predict the withdrawn amount on 2009-01-12.
3. Use the data from 2008-01-01 till 2009-01-12 to train the model.
4. Predict the withdrawn amount on 2009-01-13.
5. ...proceed analogously till forecasting the value for 2009-12-31.
6. Calculate the MAE (mean absolute error) of all 353 forecasts.

[iv] the 'm' stands for minus, i.e., in this case minus 3 days