# Methods of machine learning

## Exercise sheet II

April 30th, 2025

In this exercise class we apply decision trees to the classical *wines data set*:

1. **Task.**
   Load the data, take a look at the data documentation and extract the attributes *alcohol*, *malic acid*, and *color intensity* as features for our further analysis.

2. **Task.**
   Split the data into training and test sets (80-20 split). Train a (full) decision tree classifier using `DecisionTreeClassifier` and the Gini index for measuring the impurity. Display the learned classifier and estimate its generalization error (w.r.t. 0-1 loss).

3. **Task.**
   Now prune the learned tree. Use `cost_complexity_pruning_path` to display the resulting impurity in the pruned trees. Display the decision tree for the highest and second highest effective $\alpha$ value.

   In order to choose the best pruned tree, study the empirical and test error/risk of the pruned tree for $\alpha \in \{0, 0.002, 0.004, 0.006, \ldots, 0.998, 0.1\}$. Display the decision tree(s) with the highest test accuracy.

4. **Task.**
   Perform a grid search to find the best combination of criterion for impurity, maximum number of features, and effective $\alpha$ value. Use `GridSearchCV` from `sklearn.model_selection` and the accuracy for evaluation. Train a decision tree for the found best combination and estimate its generalization error.

5. **Task.**
   Now increase the number of features starting with the chosen three features *alcohol*, *malic acid*, and *color intensity* and adding the other available features one by one. Again, search each time the best combination of criterion for impurity, maximum number of features, and effective $\alpha$ value. Report the resulting estimates for the generalization error versus the number of features.

   Then, study the importance of all features (based on the final tree using all features in the training data set). Reorder the features according the their estimated importance and redo the above task, i.e., add now the features one by one, perform grid search, and report the estimates for the generalization error versus the number of sorted features.