

### Exercise 1: Recap Nested Resampling

Assume we have a dataset  $\mathcal{D} = ((\mathbf{x}^{(1)}, y^{(1)}), \dots, (\mathbf{x}^{(n)}, y^{(n)}))$  with  $n$  observations of a continuous target variable  $y$  and  $p$  features  $x_1, \dots, x_p$ . We want to build a prediction model that can be deployed and we want to estimate the corresponding generalization error. For this, we build a graph learner that consists of a neural network in one arm and a random forest in the other arm. The neural network shall have one hyperparameter, the number of hidden layers; assume the number of nodes per hidden layer and all other possible hyperparameters are fixed. The random forest shall have two hyperparameters, the maximal depth and the number of trees; assume that all other possible hyperparameters are fixed. In total, we pursue three goals (not necessarily in this order):

- A) Train a final model  $\hat{f}$  that can be deployed.
- B) Tune the graph learner.
- C) Estimate the generalization error.

Answer the following questions:

- 1) For each goal:
  - a) Do we need resampling, nested resampling, or no resampling?
  - b) Which fraction of the available dataset can be used?
- 2) In which order (e.g., "A-B-C") can the three goals be tackled?
- 3) Write down a pseudo-algorithm for carrying out all three steps (in a sensible order as derived in 2))
- 4) Assume the number of hidden layers is  $\in \{1, 2, 3, 4, 5\}$ , the number of trees is  $\in \{10, 50, 100, 200\}$  and the maximal depth is  $\in \{2, 3, 4, 5\}$ . Use 3-fold cross-validation as outer resampling and 4-fold cross-validation as inner resampling. Use grid search and consider all possible hyperparameter combinations. Compute the total number of model trainings carried out in 3).