**Exercise 1: Recap Nested Resampling**

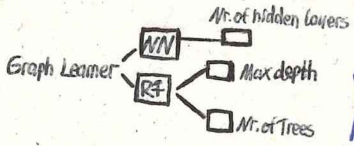
Assume we have a dataset $\mathcal{D} = \left(\left(\mathbf{x}^{(1)}, y^{(1)}\right), \ldots, \left(\mathbf{x}^{(n)}, y^{(n)}\right)\right)$ with $n$ observations of a continuous target variable $y$ and $p$ features $x_1, \ldots, x_p$. We want to build a prediction model that can be deployed and we want 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.

Simple Graph Learner
Learner-type is kind of treated as another upstream Hyperparam.




$\Lambda = \Lambda_{NN} \vee \Lambda_{RF}$
$|\Lambda| = 5 + 4 \cdot 4 = 21$

Answer the following questions:

4) Per Outer Fold: $|\Lambda| \cdot \underset{(inner)}{4} + \underset{(final\text{-}train)}{1} = 21 \cdot 4 + 1 = 85$ model fits

across three outer loops: $3 \cdot 85 = 255$

1) For each goal:

   a) Do we need resampling, nested resampling, or no resampling?

   b) Which fraction of the available dataset can be used?

Final Tuning on full $\mathcal{D}$: 85 model fits, last one is final $\hat{f}$ trained on $|\mathcal{D}|$

Overall we need $256 + 85 = 340$ total model fits

Make computation fast with Parallelization

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).

A) Train final learner $\hat{f}$

a) no resampling b) use full dataset $\mathcal{D}$ for final fit

B) Tune the graph learner

a) Use resampling to estimate $\widehat{GE}$ of different HPs and choose best $\lambda^*$

b) Use all data, but with repeated train-test splits (here potentially other splits for each graph branch)

C) Estimate GE of the whole Tuned Learning Algorithm Process

a) Nested Resampling for unbiased GE estimation, outer loop for eval. to measure $\widehat{GE}$ (average across all $\lambda^*_s$)
inner loop is for Tuning and refers to B

b) Use all data, but two-level nested train-test-splits, one outer & one inner resampling

2) Order

i) C (with B inside each outer fold)
↓
B (on full dataset)
↓ classical way
A

ii) B (on full dataset)
↓ A → C (would need fresh test data, otherwise eval on Train Error, likely too optimistic)
↓ C (Biased!) → A (Can do this if unbiased GE-estim. not necessary, e.g. learner-type already chosen by superiors)

3) Pseudo Code

Input data $\mathcal{D}$, inducer $\mathcal{I}: \mathbb{D} \times \Lambda \to \mathcal{H}$ of $f$, perf. measure $\rho$, HP search space $\tilde{\Lambda}$

split $\mathcal{D}$ into $(\mathcal{D}_{train}^{(t)}, \mathcal{D}_{test}^{(t)})$ $t = 1, 2, 3$ times {3-fold CV in outer resampling}

for $t$:

split $\mathcal{D}_{train}$ into $(\mathcal{D}_{train}^{(t,k)}, \mathcal{D}_{valid}^{(t,k)})$ $k = 1, 2, 3, 4$ times {4-fold CV inner resampling}

Foreach $\lambda \in \tilde{\Lambda}$: Train $\mathcal{I}(\mathcal{D}_{train}^{(t,k)}, \lambda)$ and measure inner performance on $\mathcal{D}_{valid}^{(t,k)}$
[grid search] (for each k) Calc. $c(\lambda) = \frac{1}{4} \sum_{k=1}^{4} \rho\left(y_{valid}^{(t,k)}, \mathcal{I}(\mathcal{D}_{train}^{(t,k)}, \lambda)(x_{valid}^{(t,k)})\right)$

End for loop, select $\hat{\lambda}^{(t)} = \arg\min_{\lambda} c(\lambda)$

• Refit $\hat{\lambda}$-config on all $\mathcal{D}_{train}^{(t)} \Rightarrow \hat{f}_t = \mathcal{I}(\mathcal{D}_{train}^{(t)}, \hat{\lambda}^{(t)})$

Do outer test evaluation $\widehat{GE}_{outer}^{(t)} = \rho(y_{test}^{(t)}, \hat{f}_t(x_{test}^{(t)}))$

End for loop, Estimate overall Generalization Error $\widehat{GE} = \frac{1}{3} \sum_{t=1}^{3} \widehat{GE}_{outer}^{(t)}$ ⟨Goal C⟩

\# Tune on full dataset (same grid search, 4-fold CV on $\mathcal{D}$) yielding us minimal $\hat{\lambda}^*$ ⟨Goal B⟩

Final Fit: Train once on 100% of $\mathcal{D}$!
$\hat{f} = \mathcal{I}(\mathcal{D}, \hat{\lambda}^*)$ ⟨Goal A⟩

\#