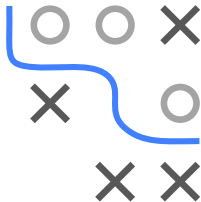


Introduction to Machine Learning

Hyperparameter Tuning In a Nutshell

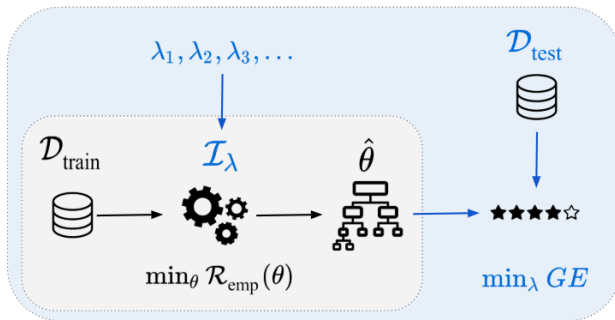
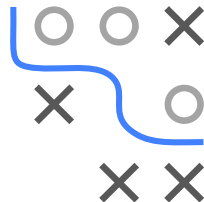


Learning goals

- Understand the main idea behind tuning,
- fulfilling the untouched-test set principle via nested resampling,
- and pipelines

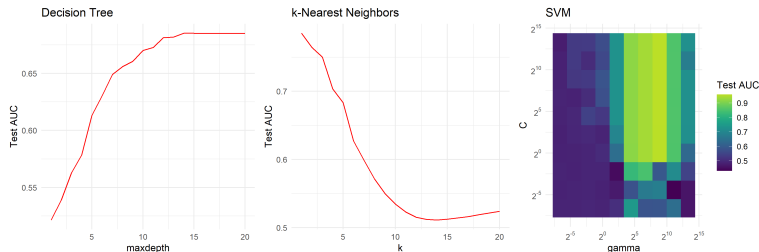
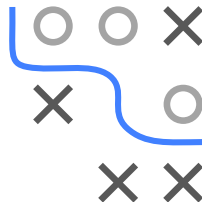
WHAT IS TUNING?

- Tuning is the process of selecting the best hyperparameters, denoted as λ , for a machine learning model.
- Hyperparameters are the parameters of the learner (versus model parameters θ).
- Consider a guitar analogy: Hyperparameters are akin to the tuning pegs. Learning the best parameters $\hat{\theta}$ – playing the guitar – is a separate process that depends on tuning.



WHY TUNING MATTERS

- Just like a guitar won't perform well when out-of-tune, properly tuning a learner can drastically improve the resulting model performance.
- Tuning helps find a balance between underfitting and overfitting.

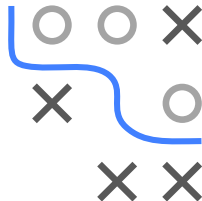


Comparing AUCs of different values for hyperparameters *maxdepth*, *k*, *gamma*, and *C*

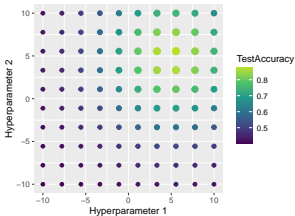
NAÏVE APPROACHES

Goal: Find a best configuration $\lambda^* \in \arg \min_{\lambda \in \Lambda} \widehat{\text{GE}}(\mathcal{I}, \rho, \lambda)$

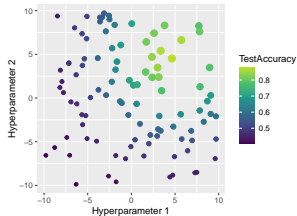
⇒ Tuners τ , e.g., **Grid Search** and **Random Search**, output a λ^*



Grid Search



Random Search

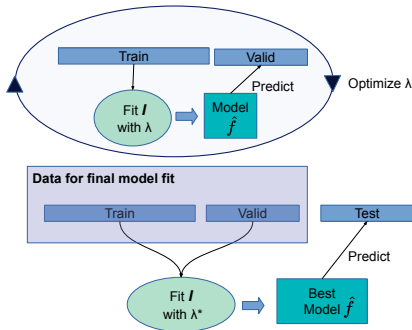
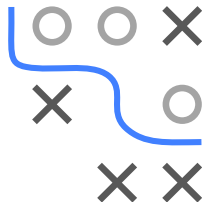


Sophisticated techniques, based on assumptions about the objective function, search for optimal solutions more efficiently.

UNTOUCHED-TEST-SET PRINCIPLE

We've found a $\lambda^* \in \Lambda$. How well does it perform?

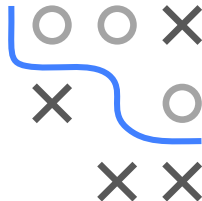
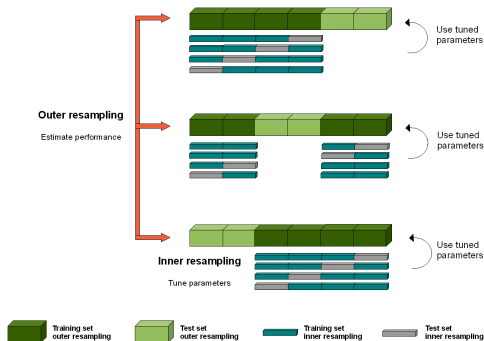
- **Careful:** We cannot use the same data for both tuning and performance estimation, as this would lead to (optimistically) biased performance estimates!
- To obtain an unbiased \widehat{GE} , we need an **untouched** test set:



NESTED RESAMPLING

To decrease variance of the \widehat{GE} , **Nested Resampling** is used:

- Just as we generalized holdout splitting to resampling, we generalize the three-way split to nested resampling (as we first have to find λ^*):



PIPELINES IN MACHINE LEARNING

Pipelines are like the assembly lines in machine learning. They automate the sequence of data processing and model building tasks.

Why Pipelines Matter:

- **Streamlined Workflow:** Automates the flow from data preprocessing to model training.
- **Reproducibility:** Ensures that results can be reproduced consistently.
- **Error Reduction:** Minimizes the chance of human errors in the model building process.

