Introduction to Machine Learning
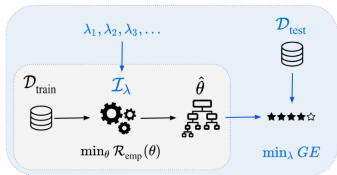
Hyperparameter Tuning
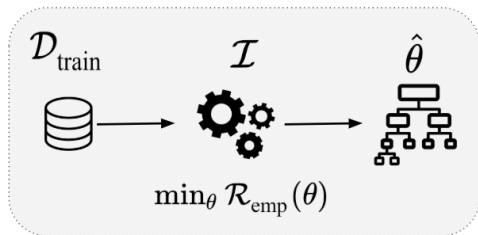# Introduction

Learning goals

- Understand the difference between model parameters and hyperparameters
- Know different types of hyperparameters
- Be able to explain the goal of hyperparameter tuning

MOTIVATING EXAMPLE

- Given a data set, we want to train a classification tree.
- We feel that a maximum tree depth of 4 has worked out well for us previously, so we decide to set this hyperparameter to 4.
- The learner ("inducer") $\mathcal{I}$ takes the input data, internally performs **empirical risk minimization**, and returns a fitted tree model $\hat{f}(\mathbf{x}) = f(\mathbf{x}, \hat{\theta})$ of at most depth $\lambda = 4$ that minimizes empirical risk.

MODEL PARAMETERS VS. HYPERPARAMETERS

It is critical to understand the difference between model parameters and hyperparameters.

Model parameters θ are optimized during training. They are an **output** of the training.

Examples:

- The splits and terminal node constants of a tree learner
- Coefficients θ of a linear model $f(\mathbf{x}) = \theta^\top \mathbf{x}$

TYPES OF HYPERPARAMETERS

We summarize all hyperparameters we want to tune in a vector $\boldsymbol{\lambda} \in \Lambda$ of (possibly) mixed type. HPs can have different types:

- Real-valued parameters, e.g.:
    - Minimal error improvement in a tree to accept a split
    - Bandwidths of the kernel density estimates for Naive Bayes
- Integer parameters, e.g.:
    - Neighborhood size k for k-NN
    - *mtry* in a random forest
- Categorical parameters, e.g.:
    - Which split criterion for classification trees?
    - Which distance measure for k-NN?

Hyperparameters are often **hierarchically dependent** on each other, e.g., *if* we use a kernel-density estimate for Naive Bayes, what is its width?