

# Exercise 8 – CART

## Introduction to Machine Learning

*Hint: Useful libraries*

### R

```
library(mlr3verse)
library(rattle)
```

### Python

```
import numpy as np
```

### Exercise 1: Splitting criteria

#### Learning goals

- 1) Perform split computation with pen and paper
- 2) Derive optimal constant predictor for regression under L2 loss

Consider the following dataset:

$i$	$x^{(i)}$	$y^{(i)}$
1	1.0	1.0
2	2.0	1.0
3	7.0	0.5
4	10.0	10.0
5	20.0	11.0

---

Compute the first split point the CART algorithm would find for each data set (with pen and paper or in R, resp. Python). Use mean squared error (MSE) to assess the empirical risk.

---

Derive the optimal constant predictor for a node  $\mathcal{N}$  when minimizing the empirical risk under  $L2$  loss and explain why this is equivalent to minimizing variance impurity.

---

Explain why performing further splits can never result in a higher overall risk with  $L2$  loss as a splitting criterion.

### Solution

The variance of a subset of the observations in a node cannot be higher than the variance of the entire node, so it's not possible to make the tree worse (w.r.t. training error) by introducing a further split.

## Exercise 2: Splitting criteria

Learning goals

Understand the effect of CART hyperparameters

In this exercise, we will have a look at two of the most important CART hyperparameters, i.e., design choices exogenous to training. Both `minsplit` and `maxdepth` influence the number of input space partitions the CART will perform.

---

How do you expect the number of splits to affect the model fit and generalization performance?

---

Using `mlr3`, fit a regression tree learner (`regr.rpart`) to the `bike_sharing` task (omitting the `date` feature) for

- `maxdepth`  $\in \{2, 4, 8\}$  with `minsplit` = 2
- `minsplit`  $\in \{5, 1000, 10000\}$  with `maxdepth` = 20

What do you observe?

---

Which of the two options should we use to control tree appearance?

### Exercise 3: Impurity reduction

Only for lecture group A

#### Learning goals

1. Develop intuition for use of Brier score in classification trees
2. Reason about distribution and expectations of random variables
3. Handle computations involving expectations

#### TLDR;

This exercise is rather involved and requires some knowledge of probability theory. Main take-away (besides training proof-type questions): *In constructing CART with minimal Gini impurity, we minimize the expected rate of misclassification across the training data.*

We will now build some intuition for the Brier score / Gini impurity as a splitting criterion by showing that it is equal to the expected MCE of the resulting node.

The fractions of the classes  $k = 1, \dots, g$  in node  $\mathcal{N}$  of a decision tree are  $\pi_1^{(\mathcal{N})}, \dots, \pi_g^{(\mathcal{N})}$ , where

$$\pi_k^{(\mathcal{N})} = \frac{1}{|\mathcal{N}|} \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{N}} [y^{(i)} = k].$$

For an expression that holds in expectation over arbitrary data, we need to introduce stochasticity. Assume we replace the (deterministic) classification rule in node  $\mathcal{N}$

$$\hat{k} \mid \mathcal{N} = \arg \max_k \pi_k^{(\mathcal{N})}$$

by a randomizing rule

$$\hat{k} \sim \text{Cat} \left( \pi_1^{(\mathcal{N})}, \dots, \pi_g^{(\mathcal{N})} \right),$$

in which we draw the classes from the categorical distribution of their estimated probabilities (i.e., class  $k$  is predicted with probability  $\pi_k^{(\mathcal{N})}$ ).

---

Explain the difference between the deterministic and the randomized classification rule.

---

Using the randomized rule, compute the expected MCE in node  $\mathcal{N}$  that contains  $n$  random training samples. What do you notice?