Machine Learning for Data Science 1

(lecture notes, only for internal use)

Blaž Zupan, Erik Štrumbelj

February 23, 2020

Contents

Chapter 1

Trees and Forests

We introduce learning via recursive partitioning of the input variable space. Depending on the learning taks, the algorithm used is classification tree and regression trees, respectively. We then describe the upgrade of the tree-learning approach with construction of a set of trees, random forests, and, as part of the latter, we introduce a more general approach of bootstrap aggregation (bagging).

1.1 Classification and Regression Trees (CART)

Basic Idea

The (generalized) linear modelling paradigm, as introduced in the next lecture, assumes that the data generating process can be interpreted with a family of distributions whose parameters are in a (transformed) linear relationship with the input variables. These are parametric models. Now we will introduce a fundamentally different modelling paradigm. One that assumes that the data generating process can be interpreted as a partition of the input variable space into homogeneous (pure) regions – regions where there is little or no uncertainty left about the target variable. For regression, the target variable for the data instances within this region is almost constant (see Fig. 1.1). For classification, a majority of data instanes in the region have the same value of the target variable.

The CART algorithm

Finding the optimal partitioning of the input variable space is in general NP-complete, even if using axis-parallel splits only. That is, it is infeasible to check all possible partitions. Instead, we will consider a greedy algorithm (CART) that is based on binary recursive partitioning of the input space, at each step choosing the best possible split (according to some pre-selected criterion). Notice that this algorithm does not use any look-ahead, and while such algorithms

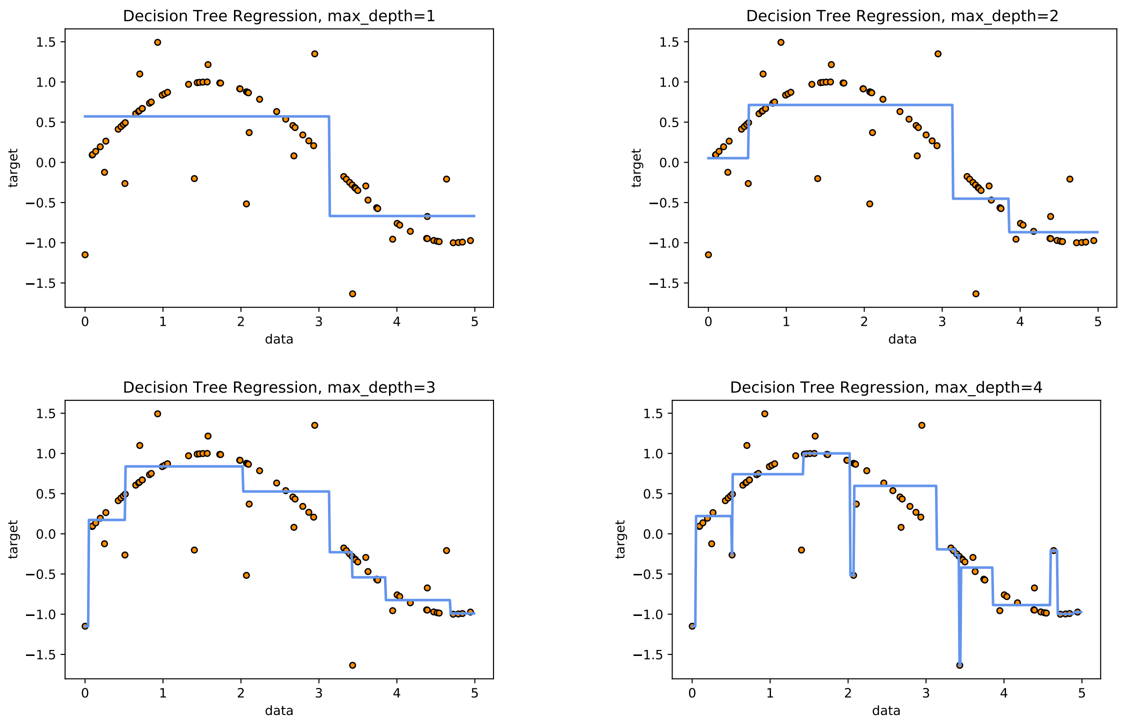

Figure 1.1: Regression trees fitted on data generated by a sine function with some noise. While the tree adapts well to the training data, its ability to overfit the training data is visible already with trees with of maximum depth of 4 (lower right).

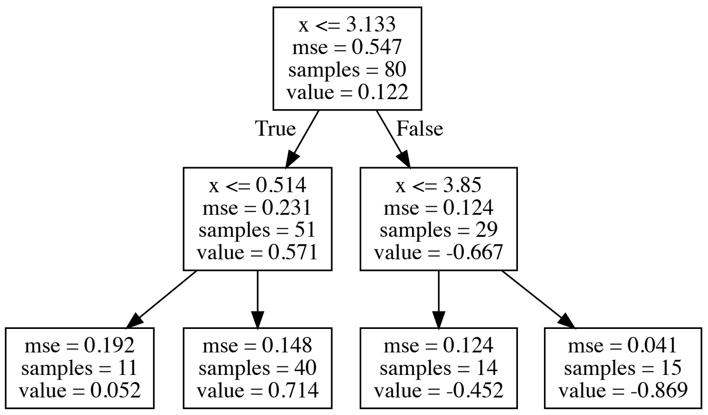

Figure 1.2: A regression tree with maximum depth of 2 from the data from Fig. 1.1.

were studied in the literature, they are not used in practice. A simplified and abstracted CART algorithm is encoded as Algorithm 1.

Algorithm 1 CART

```
1: procedure FITTREE(data)
2:     $(dataL, dataR, criterion) \leftarrow split(data)$
3:     $node \leftarrow createNode(criterion, data)$
4:     if (stoppingCriterionMet(...)) then return node
5:     $node.L \leftarrow fitTree(dataL)$
6:     $node.R \leftarrow fitTree(dataR)$
7:     return node
```

The CART algorithm uses several functions that require explanation:

- *createNode()*: This function creates an object that represents a tree node, which essentially stores the criterion on which the data in the node is split, and a possible reference to the data instances that are pertinent to the node. If a suitable node split is found, the node stores the information on its to siblings. Note that, as introduced above, the CART algorithm would construct binary trees.
- *split()*: The assumption here is that features are numerical or at least ordinal. We order every feature based on possible splits (based on unique values in the data, so we have a finite number of possible splits). And then we basically go through all possible feature-split combinations to find the one that is optimal according to our splitting criterion - the one that minimizes the sum of the cost of the left and right subtrees. Possible splitting criterions are discussed below.
- *stoppingCriterionMet()*: The stopping criterion can be one or more of the following:
  - When the partition is sufficiently homogeneous/pure. In particular, there is no point in splitting further if we have perfect homogeneity (all observations have the same value).
  - When the gain (relative to stopping criterion) is smaller than some pre-determined threshold.
  - Reaching pre-determined maximum tree depth.
  - Reaching pre-determined minimum number of obvervations in a leaf.

Choice of the Splitting Criterion