Applying Decision Tree Algorithm on the MNIST Dataset

July 21, 2021

1 Introduction

MNIST ("Modified National Institute of Standards and Technology") is the de facto "hello world" dataset of computer vision. Since its release in 1999, this classic dataset of handwritten images has served as the basis for bench-marking classification algorithms. As new machine learning techniques emerge, MNIST remains a reliable resource for researchers and learners alike.

Decision trees can be applied to both regression and classification problems. The original MNIST dataset contains 50000 digits in the training dataset and 10000 digits in the test dataset. Also there is a rotated version of the data points, and it contains 10000 digits for each of the training and testing set. Each of the sets contain handwritten image of some digit in the form of 28×28 pixel gray-scale images.

At first we will apply decision tree algorithm on the original dataset and we will observe the accuracy achieved by this model on the test data of the original dataset as well as of the rotated dataset. Then we will merge the two data set together and construct a new model and test its accuracy on the merged test set.

2 Overview

Unlike regression trees, classification trees are used to predict quantitative variables where we predict that each observation belongs to the most commonly occurring class of training observations in the region to which it belongs. As we are going to assign an observation in a given region to the most commonly occurring class of training observations in that region, the classification error rate is simply the fraction of the training observations in that region that do not belong to the most common class:

$$E = 1 - max_k(\hat{p}_{mk})$$

where $\hat{p}_{mk}$ represents the proportion of training observations in the m-th region that are from the k-th class. We can also use Gini Index as a measure of node purity to evaluate the quality of a particular split,

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}).$$

An alternative to the Gini index is entropy, given by

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}.$$

Overfitting is a significant practical difficulty for decision tree models and many other predictive models. To overcome this, we can implement the method of cost complexity pruning with decision tree which provides an option to control the size of a tree.

Pruning is not the only solution to the problem of overfitting. A natural way to reduce the variance and hence increase the prediction accuracy of a statistical learning method is to take many training sets from the population, build a separate prediction model using each training set, and average the resulting predictions. This is exactly what is done in the method of bagging; we build a number of decision trees on bootstrapped training samples. Random forests provide an improvement over bagged trees by way of a small tweak that decorrelates the trees. Each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors. The split is allowed to use only one of those m predictors. We usually select $m \approx \sqrt{p}$, i.e. the number of predictors considered at each split is approximately equal to the square root of the total number of predictors.