Predictive Modeling Report

Machine Learning Assignment - Yesmine Hachana

Introduction

The objective of this project is to build the best possible predictive model for a continuous target variable using a tabular dataset containing a mix of numerical and categorical explanatory variables. The task is a regression problem.

The dataset is provided in two parts. We have a core dataset (dataset.csv) for which the target is known to us. We also have an evaluation dataset (evaluation.csv) for which predictions must be delivered by us.

For the modeling process, I followed the following structure. First, I wrote a script used for data preparation and splitting. Then, I wrote a script for model building and selection. Finally, I wrote a script whose role is the final evaluation and diagnostic analysis.

Data Description

Structure of the Dataset

The core dataset contains 4819 observations in total, 25 explanatory variables, one identifier variable idx, and one continuous target variable target.

In the first part of the script the dataset was randomly split into a learning set with 3855 observations, and a test set with 964 observations.

The test set was used only for the final estimation of future predictive performance.

Model Building and Selection

Baseline Model

My first step was to train a simple decision tree regressor with default parameters. I used a 80/20 split. The baseline RMSE is 0.9152, which is a high error.

Resampling Strategy

To perform model selection, I used in the second part of the script a 5-fold cross-validation. The same folds are used across all competing model. I evaluated to performance by using the Root Mean Squared Error (RMSE).

Tuning the Hyperparameters

I built two machine learning algorithms. The first one is a decision tree regressor. For this decision tree, I explored several parameters. I explored a max depth of 3, 5, 8, 12, or 14. Regarding min samples split, I tested 2, 10, 50, 100, and 200. For this regressor, I found the best configuration to be a max depth of 12 and a min samples split of 100. For the cross-validated results, the mean CV RMSE was 0.7146 and the standard deviation 0.0212.

The second algorithm I build was a random forest regressor. I also explored several hyperparameters for this algorithm, namely n estimators (200, 300, and 400), max depth (2 and none), and min samples split (2, 5, and 10). The best configuration was n_estimators = 300, max_depth = none, and min_samples_split = 2. Regarding the cross-validated results, the mean CV RMSE was 0.6208 and the standard deviation 0.0169.

Final Model Selection

The random forest outperforms the decision tree on cross-validated RMSE. So the final model I chose is the random forest regressor. I retrained it on the full learning set.

Expected Performance on Future Data

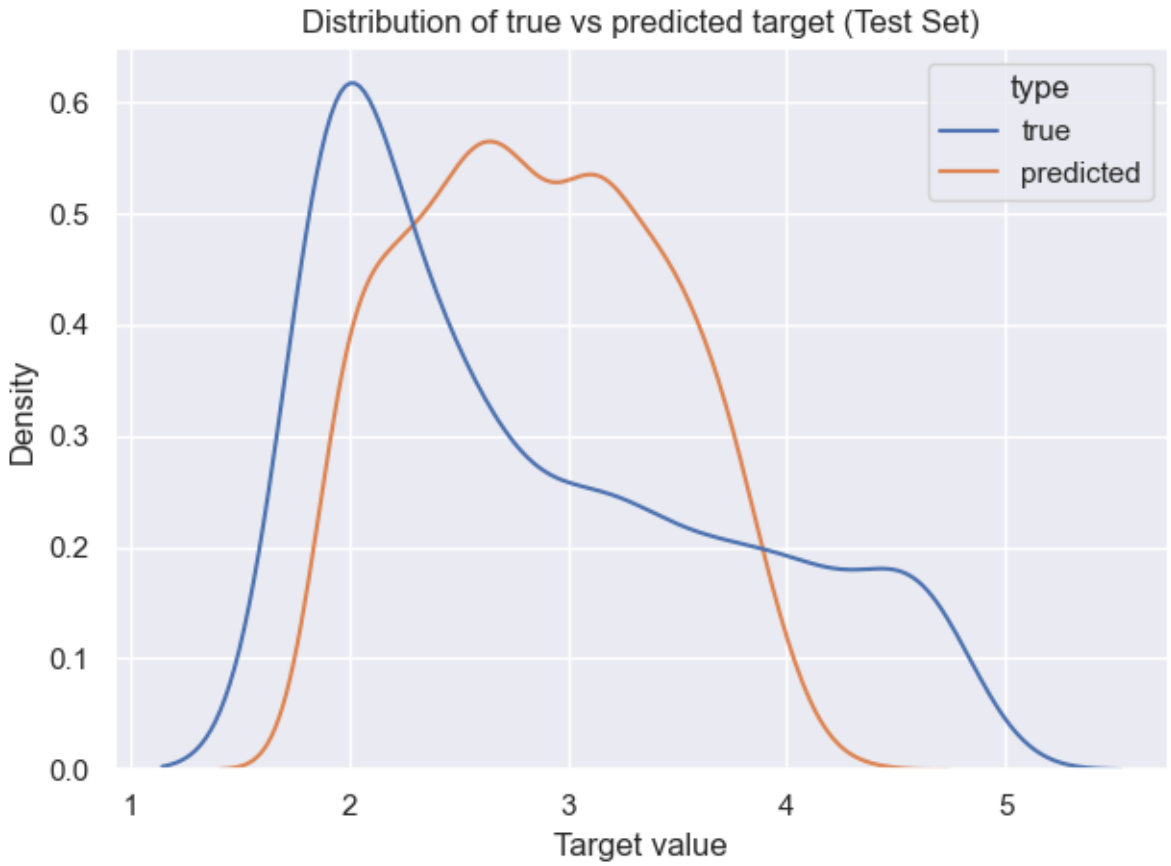
Here are the final results I found on the test set:

- An RMSE of 0.6351
- A MAE of 0.4992
- An R Squared of 0.5115
- A Median Absolute Error of 0.4223
- A 95% Error Interval between -1.4354 and 0.9700
- A Maximum Absolute Error: 2.2732

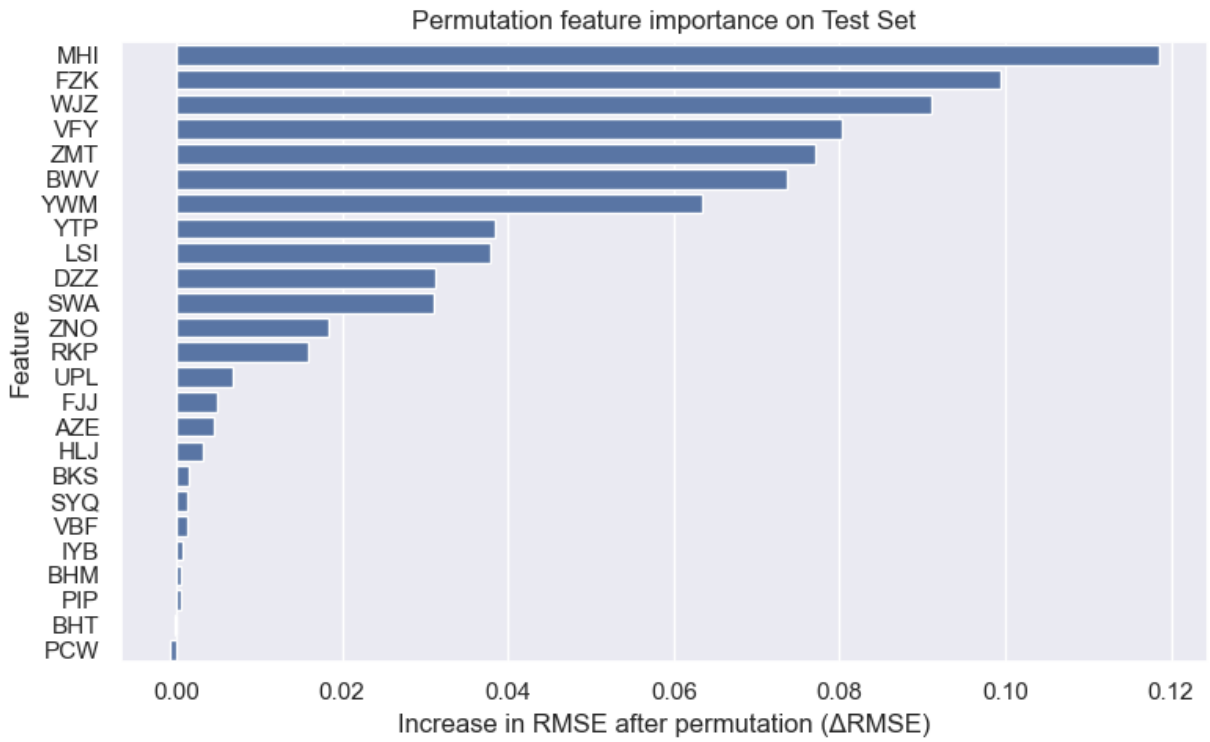
The fact the values are close between CV RMSE (0.6208) and Test RMSE (0.6351) may tell us that the model generalises well or does not overfit.

The scatter plot shows that the model does take into account the increasing relationship between true and predicted values. But there is some dispersion around the diagonal.

It seems like errors are centered quite close to zero with a bit of asymmetry and a heavier negative tail. Most errors lie between -1 and 1 though.



The predicted distribution is more concentrated. It has a weaker right tail than the true distribution, maybe because extreme values are harder to predict.



I calculated the permutation importance by randomly permuting each feature and measuring the increase in test RMSE. As said in class, this only shows predictive importance and not causal effects.

Finally, the largest absolute error observed on the test set is 2.27 and is linked to an observation with high numerical values on several explanatory variables.