AI Hack Submission - Boston Housing Dataset

February 2021

I. Introduction

Our goal was to achieve an accurate prediction for the MEDV using a machine learning model.

II. Data Engineering and Processing

We inspected the data using python to determine whether the data needed imputing. We found that data had not been omitted and therefore imputation was not needed.

III. Methodology

We loaded the data using `pandas` and plotted it using `matplotlib` to perform an initial inspection of the data.

We plotted each feature against the MEDV to determine if any primary observations could give us an indication of the effect of that feature on the MEDV. We then hypothesised possible informal reasons for each relation if the context was permissible.

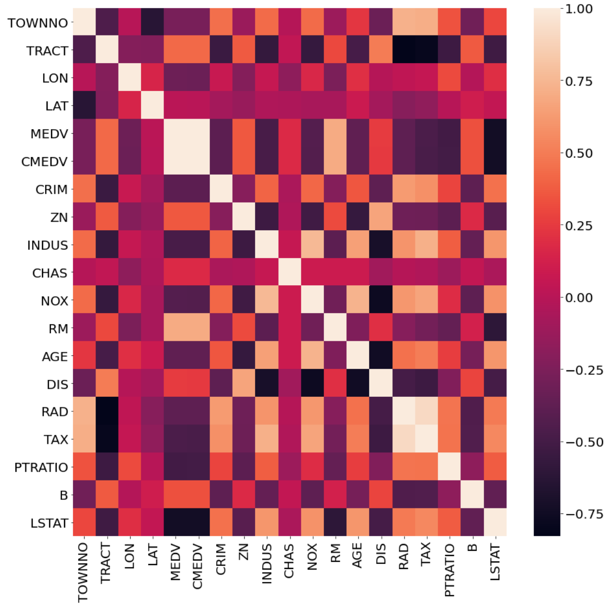

Fig. 1. Generated heatmap we used to initially inspect the data..

For instance, we considered the relationship between CRIM and the MEDV which showed a general inverse relationship. We speculated that historical observations have shown that home owners desire homes in areas with a lower crime rate, increasing the demand and therefore prices of houses in low crime rate areas.

We initially decided to use a linear regression model to fit our data. Upon careful consideration we had difficulty choosing features to be included in the model. We had no quantitative metric to assure us that a given feature would be better at predicting the MEDV than another. Therefore we could not formally justify any given choice.

We then considered a random tree regressor. The main point of advantage we determined using a random tree regressor over a linear regression model was that inclusion of all features could be included in the random tree regressor. As we could not determine the significance of one feature over another, we thought best that the model should determine the weightings itself.

IV. Model Evaluation and Optimisation

We initially evaluated our model by executing our model with different learning parameters. For a given value of a parameter, we performed 10 iterations of the model with different random seeds to evaluate the mean and standard error. Addtionally each iteration also calculated the effect of predicting the model on our test data (the proportion of learning and test data we also varied).

We plotted both the mean prediction on the test and training data and their corresponding errors against the range of values for a specific parameters.

We found evidence of overfitting given that the model performed much better on the training data than the test data.

In light of this, we used a 3-fold cross validation to optimise our regressors hyperparameters with the objective of reducing overfitting. We found the optimal set of parameters by performing a randomized search (using RandomizedSearchCV) with 100 iterations.

We then used the updated learning paramaters in our regression model and refitted the data.

V. Results

The R^2 value for our random tree regressor model is shown in figure 5. Our model was able to predict the test data well given the R^2 value. We anticipate overfitting, as we lacked enough data to ensure this was not a risk. Our optimised model fit gave an R^2 value of 0.88. We obtained higher values for R^2 before optimisation.

Apendix

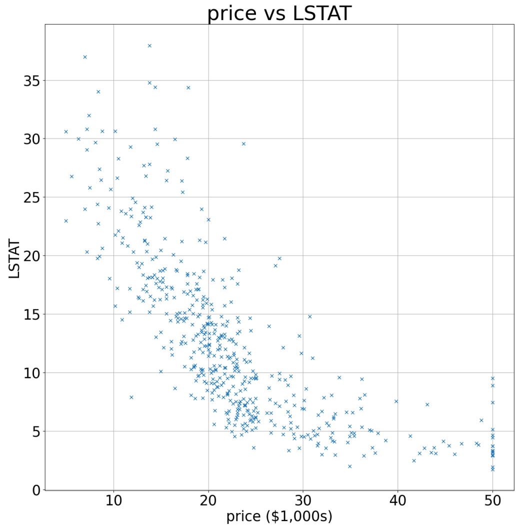

Fig. 2. Example of relationship between price and a feature. LSTAT is the percentage of lower status of the population.

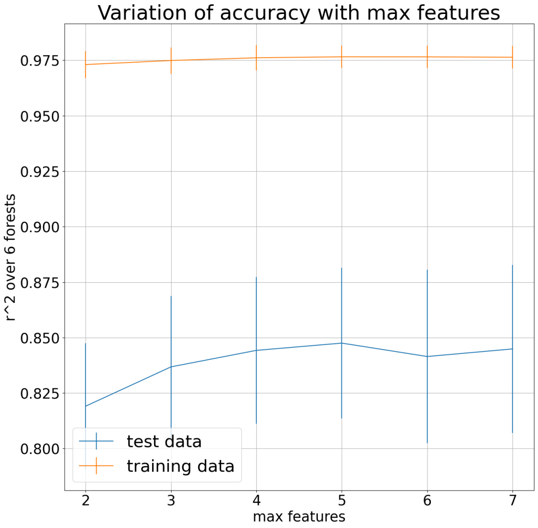

Fig. 3. Plot to illustrate how the accuracy of the AI varies with changing the max features used in each tree of the random forest. Each value for max features was repeated 10 times with 10 different seeds to determine a mean and a standard error.

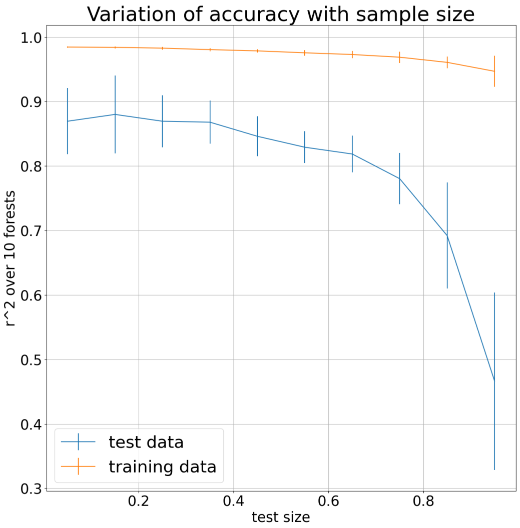

Fig. 4. Plot to illustrate how the accuracy of the AI varies with changing the proportion of test and train. Each value for the proportion was repeated 10 times with 10 different seeds to determine a mean and a standard error.

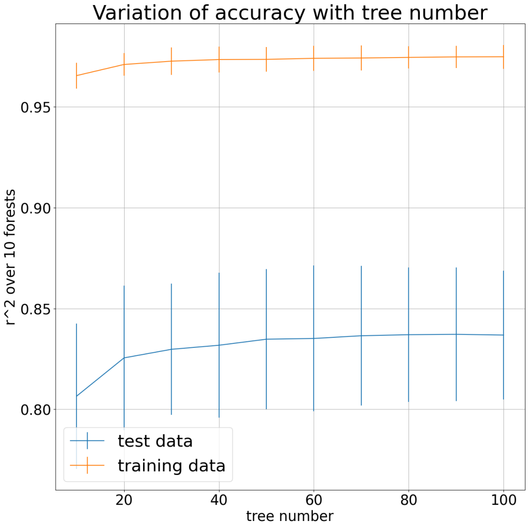

Fig. 5. Plot to illustrate how the accuracy of the AI varies with changing the number of trees in the random forest. Each value for tree number was repeated 10 times with 10 different seeds to determine a mean and a standard error.

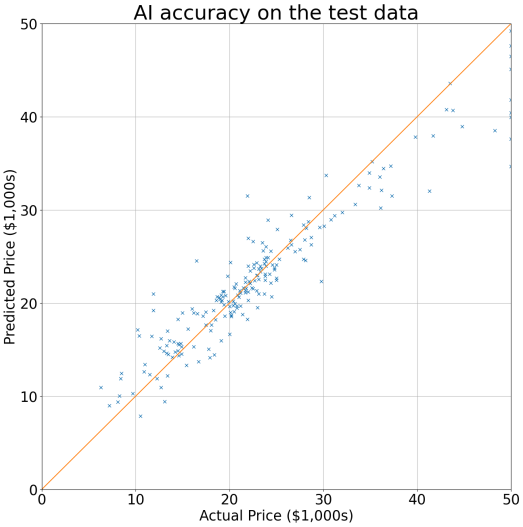

Fig. 6. Plot to illustrate how the AI predicts house price. We used the optimum parameters for generating the forest. This gave an R^2 of 0.88.