USED CAR VALUE PREDICTION PROJECT MACHINE LEARNING ANALYSIS

Dataset:
Each machine learning problem is closely related to the dataset and the problem-solving phase starts with a clean dataset and good understanding of the weight of those parameters in building up the model.

The dataset was not prepared by professionals and a rigorous clean-up process had to be carried out to minimize the loss of data and to zero out faulty data feed to the prediction models. The observations being quite arbitrary, coupled with great percentage of missing data posed great difficulty. Missing values, most of which could be normally completed with interpolation or other conventional fill methods had to be picked out deleted, since all were non-fillable missing values. This resulted in loss of approximately %38 percent of the original data. However, despite the spread and diversity of the data, adequate data fidelity was achieved and the extensive data wrangling efforts paid off with satisfactory model outputs.

Understanding the Problem:
Dealing with used car values ranging from 200€ to 1.25M € built from 1950's to 2016 was not going to be easy for any machine learning model. The model had to process data that has a price standard deviation of 10558€ and a mean price of 6894€. There are many brands that have similar features but a wide range of price. Initial thought was to implement a linear regression model based on Scikit learn, however the results were not satisfactory, since data consists of both continuous variables and classifiers and linear models cannot have classifiers as input and simply have to ignore them. In order to better model this diverse dataset, next approach was to utilize regression tree models, again from Scikit learn.

The decision tree models proved to be far more successful from the start since they were able to capture non-linear relationships between features and the target: prices, following these steps:

- Dummy variables were created so that categorical features could be inserted to the model.
- Data was split to various test fractions from %20 to %40 by train test split.
- Model was instantiated and then fit on training data.
- Cross-validation of 3-10 folds were applied, depending on processing time.
- Fitted model predicted test set values, the unseen part of the dataset.
- Grid-search and Randomized search and hyper parameter tuning was carried out, processing power and time permitting.
- Sklearn metrics R^2 and RMSE were used for scoring.

Based on the general guidelines above, numerous models including DecisionTreeRegressor, RandomForestRegressor, BaggingRegressor, GradientBoost, Adaboost and Ridge were evaluated. In all of those models, one drawback was immediately identified: the test size was dominant on the quality of the model, which was a strong indicator of some observations being too outlying for the model to explain. Basically, the validation performance or the test accuracy of the model was heavily dependent on where outlying data was. Test score turned out to be higher than train score, which clearly indicated a problem.

Initial reaction was to revisit the data frame and filter out vehicles older than year 1990 and more expensive than 400.000€. However, these surprisingly did not change the nature of error. Still, data was sensitive to folding and train test split, only the amplitude was reduced. With further pruning and sacrificing the outlying data points, train test score anomaly was fixed, and after many hyperparameter tuning trials the results' value of 2281€ and test score of 0.917 was achieved with a RandomForestRegressor, which appeared to be the second most successful

model so far, surpassed only by GradientBoostRegressor that produced slightly better results:2232€ and 0.920.

Even though the R^2 score of the test data was good, the RMSE value was not satisfactory since it was too large. In order to fix this problem, the price column (labels) of the dataset was transformed to logarithmic scale and the prediction models were re-run. Also, in addition to previous models, LightGBM(LGB) and CatBoostRegressor(CB) were added. Moreover, the last stage of the data filtering were reverted, including very old and very expensive cars, increasing the spread of the data. Surprisingly, after the log transformation, the RF results turned R2 score of 0.893 and an RMSE of 1.43€. Same results for GB were 0.901, 1.417€ . Newly introduced CB and LGB proved to be performing slightly better, with score values of 0.907, 1.41€ and 0.903, 1.41€, without sacrificing any data to filtering.

The other method for normalizing the price variable, box cox power transformation, provided similar results. CB model had an R^2 score of 0.91 and RMSE score of 1.6€

Conclusion and Recommendations:

Linear models failed to provide an effective mechanism to predict the price, however decision tree based non-linear models and ensemble models together with boosting models achieved high success.

Fine hyperparameter tuning, a step that was skipped due to lack of computational power, remains as an area of improvement.