ID5059 Individual Report

Introduction

With car industry being one of the largest industries in the world, predicting the price of cars accurately is crucial. This report aims to analyze the prediction of car prices using a machine learning algorithm. The process of building and testing the models included feature selection, data cleaning, imputation, model training and model testing. The data used started with a small size(10MB), requiring a larger dataset (100MB) for better performance, both datasets are from Kaggle.

Data Preparation

Feature Selection: The small dataset initially contained columns having NAs exceeding 50%, so we moved these columns to avoid bias. We also removed numerical variables with units, as well as variables based on personal experience. Then, we conducted correlation analysis and scatter plots to examine the relationship between numerical variables and price. We used the chi-squared test to examine the relationship between categorical variables and price, and removed variables with a significant relationship to improve model efficiency and effectiveness.

Data Cleaning: NaN values in numerical attributes were replaced with the median, a robust measure of central tendency that is less sensitive to outliers. NaN values were replaced with the most frequent category for categorical attributes, deemed the best approximation.

Scaling and Label Encoding: The distribution plots indicated that variables were skewed. This non-normal distribution can lead to issues when using linear regression models. To mitigate this issue, the numerical variables were standardized. This normalization allowed the model to compare variables on an equal footing, resulting in improved model performance. In addition, categorical attributes were encoded into numerical values to allow the model to understand the relationships between different attributes. This ensured that the dataset was clean and ready for the next stage of the analysis.

Model Selection and Training

Here is a table about output from each model trained and correspondent hyperparameters. And we compute RMSE((Root Mean Squared Error) and SD (standard deviation) through cross-validation with 10 folds.

	Linear Regression	Lasso Regression	Ridge Regression	Elastic Net Regression	Decision Tree	Random Forest
Regularization term	--	L1-norm	L2-norm	L1-ration=0.5	Hyperparameters: criterion=mse splitter=bes' max_depth=None min_samples_split=2 min_samples_leaf=1 min_weight_fraction_leaf=0.0 max_features=None random_state=None	Hyperparameters: n_estimators=100 criterion=mse max_depth=None min_samples_split=2 min_samples_leaf=1 min_weight_fraction_leaf=0.0 max_features=auto max_leaf_nodes=None min_impurity_decrease=0.0 bootstrap=True oob_score=Fals n_jobs=None random_state=None
Penalty term	--	α=1	α=0.1	α=0.1		
RMSE	530	745	1958	5581	313	1634
SD	1328	893	708	616	1174	1107

Even though the Elastic Net regression model has a smaller SD compared to any other, meaning the model's performance is better when trained on different subsets of the data, scatter plots show there exists a strong non-linear relationship. The random forest model outperforming the Elastic Net model in terms of accuracy and generalizability, based on relatively lower RMSE, was chosen as the best model. Subsequently, a grid search with cross-validation is employed to optimize the model's hyperparameters. This entails specifying a range of values for the n_estimators, max_depth, and max_features parameters. Nevertheless, despite the exhaustive search, the Root Mean Squared Error (RMSE) obtained from the optimal hyperparameter combination is not substantially superior to that of the initial model. Consequently, the original model is retained.

Model Test

To assess the accuracy of the random forest model for predicting car prices, a 20% test set was used, which was held out from the original dataset. The trained model was applied to the test set, and RMSE (measures the error in the predicted values) is around 1961. The score of this model, R-squared (measures the goodness of fit of the model), is 0.983. Both results indicate the model has a high degree of correlation with the test data. It also can be seen the same pattern from the scatter plot of the actual price and predicted price. Overall, these results indicate that the Random Forest model is a reliable predictor and can be used to make accurate predictions for new car data.

Retraining and Retesting on Large Data Size

A larger dataset can help improve the performance and accuracy of your model, as it provides more diverse and representative samples. In such cases, it is advisable to rerun the machine learning pipeline by retraining and retesting the random forest model. The model was then trained on the larger dataset and evaluated using the test set. However, when trained on the larger dataset and evaluated using the test set, the model showed an unusual result with an RMSE of 7879 and an R-squared value of 0.916. Possible reasons for this include sampling error and model sensitivity. Random forest models are built on multiple decision trees, where each tree is built on a random subset of the training data. The random subsets used to build the trees in the larger dataset may not be representative of the overall population, and the model may be sensitive to the size of the dataset, leading to larger RMSE and smaller R-square values when tested on the larger dataset.

Conclusion

In conclusion, this report highlights the process of building and testing machine learning models to predict car prices. The random forest model was found to be the best model, showing high accuracy and generalizability. However, when retrained and retested on a larger dataset, the model's performance was not as expected, suggesting that random forest models may be sensitive to the size of the dataset. Therefore, it is important to understand the limitations of the models used in data analysis.