USED CAR PRICE PREDICTION

2021

Satish Agrawal, SANJAY JARAS

BELlevue university

**Abstract**

On average, there are 1.88 vehicles per USA household. Owning a car is becoming more and more common worldwide. The current market size of used car vehicles is around $89 billion. The buyer confidence will likely increase over the next five years as the economy recovers from the coronavirus pandemic. It is projected that the revenue for used car dealers in the USA will amount to approximately $123.3 billion by the year 2024[1]. In the US alone, there are 123,905 businesses involved in the used car market. Due to the increased price of new cars and customers with a lack of funds, used car sales are on a global increase. Developing countries adopted the lease culture instead of owning a new car due to affordability. Therefore, the rise in used car sales is increasing exponentially. There is an abundance of used car sales data out and available. This project aims at using the multiple attributes of the used cars sold over the years and trains a model that can predict an appropriate price for a used car. Using machine learning algorithms like linear regression, K-Neighbors Regressor, Random Forest Regressor, etc., we aim to build a model that will help in reliably predicting used car price.

**Intro/background of the problem**

The price of the new vehicle is determined by the manufacturer. The manufacturer considers a variety of factors including government taxes, used raw materials cost, the labor involved, intended profit margin per unit, and many other factors that may contribute to the price of the car to come up with the Manufacturer Suggested Retail Price (MSRP). So, buyers of new cars are a bit more confident about the price of new cars; which is not always true with the price of an old car. The used car buying is a very complex process, as an average buyer might not think of all the variables affecting or involved in the price of the vehicle. Car sellers seldom take advantage of such a scenario by listing unfair prices owing to the demand. On the seller's part also it's quite hard to estimate the used car price manually. Generally, experienced sellers can think of some of the parameters like mileage on the vehicle, condition of the vehicle, fuel type, and vehicle age, etc. However, for experienced sellers also, it is hard to consider all parameters while estimating used vehicle price. So, there is a necessity for a used car price prediction system to reliably determine the fair price of the car using various vehicle parameters. There are existing models in the market that estimate the used vehicle price; we are not confident about the accuracy and the quality of the existing models. Depending on the organization that developed these models, those may have biases to benefit the seller.

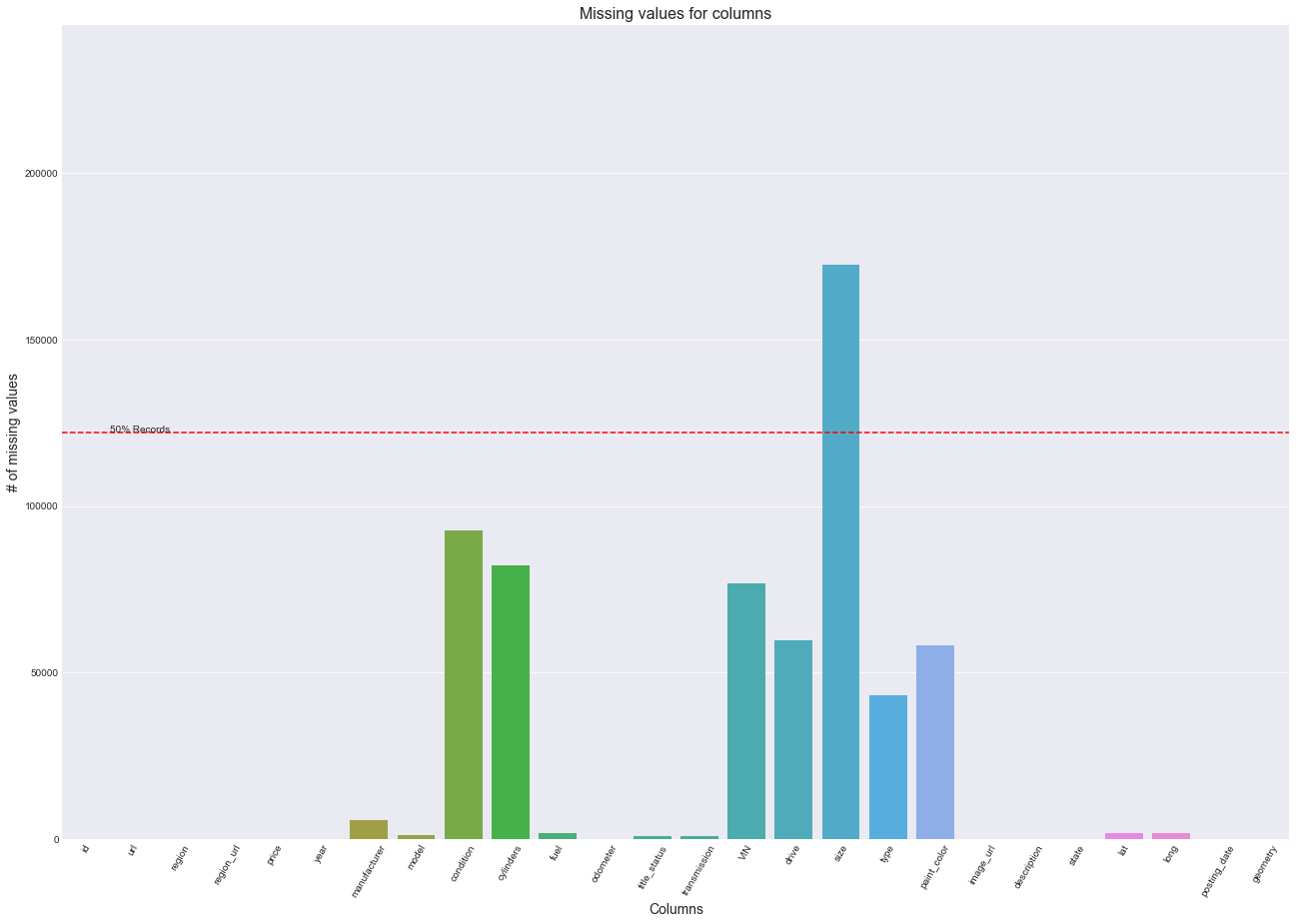
This project aims to train models for the data set chosen. The data set is one from Craigslist used car listings. We are expecting it to be generalized to the rest of the world and other listing portals.

**Methods**

We chose a data set from Kaggle @ https://www.kaggle.com/austinreese/craigslist-carstrucks-data. This data set has more than 450k records and 25 attributes for each record. We are training a supervised learning model; the target variable is the “price” of the vehicle. We have split our data into the training set and test set to avoid test data getting used in imputation and training. We are dropping a few columns from the start before starting any analysis considering those may not correlate as much (or at all) to contribute to the price of the used car. We dropped unique listing id, image URL, listing URL, region URL, VIN, and description of the car. These columns do not have a direct correlation with the target variable. As part of the data cleansing process, we looked at the distribution of the listing year column. We removed records locate outside of the United States by latitude and longitude. We restricted records only from latitude between 25 and 50, and longitude between -125 to -65. We dropped records with extreme values for the year column with a value less than 1995 and more than 2020. We also removed the outliers from the dataset based on the odometer readings and the final price. We removed records with odometer readings of more than 170,000 miles. For removing records by price, we decided to remove records with a price outside of the range of $2000 to $60,000. As part of feature engineering, we have added a new column for the age of the car at the time of listing. The value for the age in the month is derived from the model year and the date of listing. For the model year month, we considered September of the previous year of the model year as most of the car manufacturers launch their next year's car models in August/September of the current year. We found that the distribution of the age for the data set is a right-skewed distribution with a single peak.

There are many categorical attributes in the data set which are key to our modeling project for price prediction, for example, manufacturer, model, cylinders (number of cylinders), title status, transmission (auto versus manual), fuel (the type of fuel used) and few others. We used ordinal encoding to convert categorical columns into numerical columns to use with model training. We decided to use sklearn’s latest version 0.24 for label encoding. This version gracefully supports the missing labels from unseen data.

We also noticed a significant amount of missing values in a few of the columns. After analyzing the percentage of missing values, we decided to drop the “size” column from further processing and model. The “size” column has more than 50% missing values.



To fill missing values, we implemented iterative imputation using many of the estimators(BayesianRidge, DecisionTree, ExtraTrees, KNeighbours, and Lasso) to reduce the mean square error. We chose Bayesian ridge, decision tree regressor, extra trees regressor, K neighbor regressor, and lasso regression and ran through the data set for columns with missing values using the negative mean square error being the scoring metric. It came out to be the Lasso the best imputer for the data set. We finally used the chosen estimator to impute all the missing values.

We further calculated the correlation between all the dependent variables with the target variable. From these correlation numbers, we found that the “state” column has a very small correlation with the target variable. We decided to drop column “state” from our model training. As the result of feature engineering, we added the column age of the vehicle and dropped columns unique listing id, image URL, listing URL, region URL, VIN, Description, state, and size from further processing.

As the values of all the columns are on different scales we decided to use StandardScaler from sklearn to normalize the values. We created pipelines for all the above operations so that we perform the same kind of operations on the test dataset as well as the future data after model deployment.

after model deployment.

For model selection, we used an exhaustive search technique. We decided to use the following different algorithms for the exhaustive search LinearRegression, DecisionTreeRegressor, XGBRegressor, RandomForestRegressor, KNeighborsRegressor, Ridge, Lasso. Out of 400K plus training set, we use only 50K records for an exhaustive search to avoid the crashing of the program while training models for model selection. We used Sklearns’ Grid Search Cross-validation method for this purpose. This method allows us to train multiple models of the same or different algorithms with several hyperparameters. We used 5-fold Stratified cross-validations, stratified split tries to keep the same number of records of each category in every split. Grid search trains all the models with a different combination of splits, algorithms, and hyperparameters. Grid search returns the score of each model as a result. We decided to use R-square as a scoring metric. The most common interpretation of r-squared is how well the regression model fits the observed data. In the exhaustive search process, we ended up training 280 models on the training dataset(50K records). After a comparison of scores of models, we found the XGBRegressor as the best model. It has a score of 0.9061 followed by RandomforestRegressor with a 0.8722 score on the training dataset(50K records).



We decided to use XGBRegressor with the best hyperparameter found in the grid search as our final model for training and deployment. After training the XGBRegressor model on the full training dataset, we tested it with the test dataset.

**Results**

We split the data set into train and test set using an 80-20 split before starting exploratory data analysis. We want to make sure that we have some unseen data set aside for the model the see how it performs. The trained XGBRegressor model when tested with the test dataset, we saw an R2 score of 93.42%. This accuracy indicates that more than 93% of the time in the test set, the model was able to predict the correct vehicle price. This accuracy looks good from the initial analysis, we are planning to fine-tune and train other models to compare the performance.

**Discussion/conclusion – Next steps**

The initial results at this point on the project seem to have acceptable accuracy to be able to predict the price for a used car given other attributes. We have spent enough time handling missing values and outliers. Most of the attributes are now close to a normal distribution which is helping the predictability of the model. Currently, we have imputed the whole data set including the train and test set which might cause data snooping problems. We are planning to split the data set into train and test sets before imputing so that the same imputation pipeline can be easily applied to new test data in the future. We have used the Bayesian Ridge method to impute missing values in categorical attributes which is label encoded and giving decimal values for some of the records. We are working towards using simple imputer and one-hot encoding to have better imputation and accuracy in turn. We are working towards tuning hyper-parameters of the Linear regression model to come up with a better performing model. We also intend to train a few other models and do hyperparameter tuning and choose the best model for the problem. We are planning to use GridSearchCV and do an exhaustive training of multiple algorithms with multiple hyper-parameters.

**Acknowledgments**

The authors of this project have referred to Kaggle for the data set and some of the notebooks available in the Kaggle. There is a significant number of solutions available on the Internet for the problem. We have referred to some of those available solutions and have designed our own solution. We have also referred to multiple websites including data science articles from various authors, geeks for geeks, machine learning websites, mediam.com, and many others for the basic machine learning and statistics concepts and practical examples.

**References**

1. IBIS world Oct 14, 2020 - Used Car Dealers Industry in the US - [**https://www.ibisworld.com/united-states/market-research-reports/used-car-dealers-industry/**](https://www.ibisworld.com/united-states/market-research-reports/used-car-dealers-industry/)
2. Aurélien Géron – June 2019 - Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow – O’Reilly Publication
3. Panwar Abhash - Used Car Price Prediction using Machine Learning - https://towardsdatascience.com/used-car-price-prediction-using-machine-learning-e3be02d977b2
4. Used car dataset - <https://www.kaggle.com/austinreese/craigslist-carstrucks-data>
5. Enes Gokce - Predicting Used Car Prices with Machine Learning Techniques - https://towardsdatascience.com/predicting-used-car-prices-with-machine-learning-techniques-8a9d8313952