

SUMMARY

- *Overview*
- *Process*
- *Comparison - Machine Learning Model Performance*
- *Conclusions*
- *Technologies Used*

Overview

The digital era has created a shift in the automotive industry towards contactless business models. This has allowed for the car market to grow significantly larger than the new car market, primarily due to the wealth of information being offered to potential buyers online. While pre-owned vehicles tend to come with much lower price tags, there's more to consider when choosing them, which makes it difficult to accurately predict what one should be paying.

Estimating the true market value of a cars involves the daunting task of weighing several different factors. This presents the need for an accurate price prediction system. Current websites that offer this service may not comprehensively consider the actual market, which leaves buyers in a compromised position. An accurate price predictor could offer buyers this confidence, while benefitting the legitimacy of pricing websites, and assisting both car dealers and private party sellers in competitively pricing vehicles in order to quickly sell them with less hassle.

Using the wealth of car sales data already out there, Machine Learning (ML) algorithms can build a model in order to make predictions without being explicitly programmed to do so. In this study, we applied eight different ML models to a car sales data set

containing almost a 120k car listings to create a price prediction model of 93% accuracy.

ML algorithms build a model based on sample data, known as "training data", in order to make predictions or decisions. Here, 90% of the data set was split for training and the remaining 10% was used for testing the chosen algorithm. By using an extremely large data set, the algorithm becomes more experienced in how each of 7 factors influences car prices.

Process

The car sales including columns like 'Advertised_Date', 'Sold_Date', 'Sale_Price', 'Odometer', 'Power', 'Fuel_Urban', 'Make_Code', 'Family_Code', 'Year_Group', 'Region_Description', and 'Vehicle_Code'. In total, the original .csv file contains 122,595 vehicle listings.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 122595 entries, 0 to 122594
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Advertised_Date        122595 non-null object
1   Sold_Date              122595 non-null object
2   Odometer                122563 non-null float64
3   Power                  122516 non-null float64
4   Fuel_Urban             108789 non-null float64
5   Make_Code              122595 non-null object
6   Family_Code            122595 non-null object
7   Year_Group             122595 non-null int64
8   Region_Description     122595 non-null object
9   Vehicle_Code           122595 non-null int64
10  Sale_Price              122595 non-null float64
dtypes: float64(4), int64(2), object(5)
memory usage: 10.3+ MB
```

Figure 1: data information

The following steps were performed on the data set:

- Data Cleaning (Identifying Null Values, Filling-In Missing Values & Removing Outliers) Using pandas, NumPy, & seaborn
- Data Preprocessing (Standardization or Normalization) & Splitting
- Training & Testing the Data Using 8 Algorithm Models Obtained from scikit-learn, yellowbrick, & XGBoost ML Libraries in python

Since this study concerns price predictions, it's important to consider how price distributions in regression models are skewed right, as shown in the distplot diagram of the actual data set below. For any fixed value of X (independent / predictor variable), the Y value (price / dependent / target variable) prediction will be inaccurately higher than it actually is and should be corrected for. To solve this problem, a log transformation is used to scale the price, thereby helping to generate more accurate predictions of the actual target values. For this reason, evaluations of ML model accuracies are calculated based on Root Mean Squared Log Error (RMSLE) & the Coefficient of Determination (R^2).

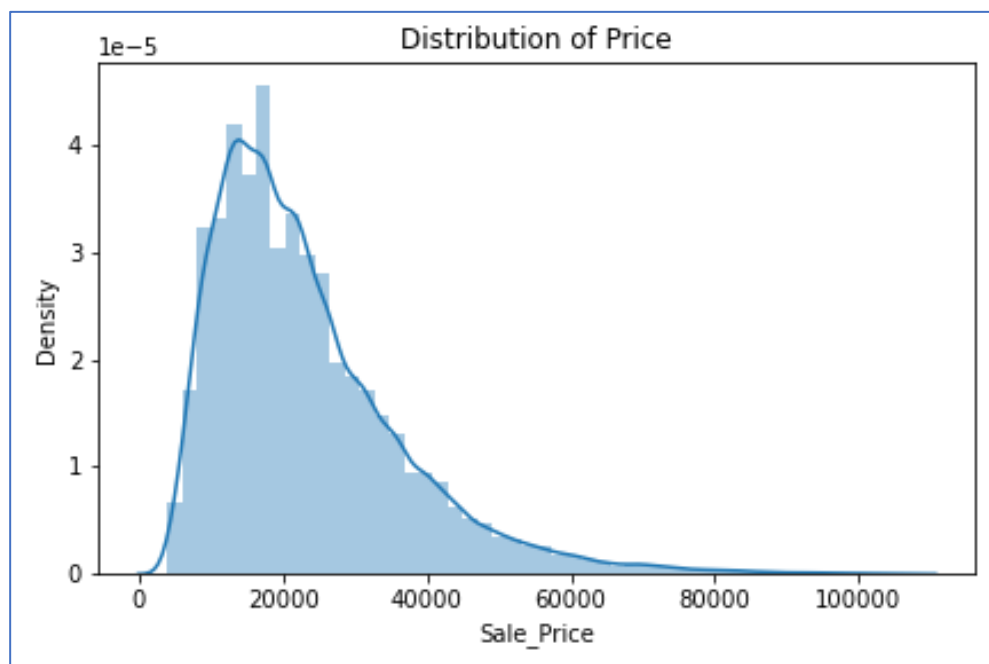


Figure 2: Distribution of price

Comparison - Machine Learning Model Performance



Figure 3: Machine Learning Model Performance comparison

Of the eight different ML models we explored, five produced prediction accuracies of 90% which is a substantial result. These algorithms were Linear Regression, Ridge, Lasso, KNN, Random Forest Regressor, Bagging Regressor, Adaboost Regressor, and XGBoost Regressor. By performing different models, we were able to learn more about the data set and the relative importance of the variables. As each ML algorithm was applied, information regarding the four statistical measures observed in the table above was gathered in order to assess the suitability of the ML model in meeting our studies objective of accurately predicting car prices.

The first three models were based in linear regression and were all found to be unsuitable for our prediction needs based on their poor MSLE, Root MSLE and R^2 scores. The relative importance of the features/variables generated by these early models provided useful insights however, since they indicated that several factors were influencing car price. As we performed more learning models on the data, a few

tendencies became apparent in that relative importance of features were always found to be most for both “year” and “odometer”, and then a handful of others.

Moving onto more complex algorithms that allowed for much larger decision trees and depth parameters is when higher accuracies of price predictions were achieved. Our data set contained 3 categorical variables and 4 numerical variables, which despite being encoded and scaled prior to utilization by the ML models, presented the need for a model that could be sensitive to the very broad importance value range for many different variables.

The most suited algorithm was found to be the XGBoost ensemble-based ML model. The advantage of the XGBoost model is its scalable and accurate implementation of gradient boosting techniques and several advanced features for model tuning, computing environments and algorithm enhancement. XGBoost can perform the three main forms of gradient boosting (Gradient Boosting (GB), Stochastic GB and Regularized GB) and is robust enough to support fine tuning and addition of regularization parameters. The 93.48% accuracy model established here provides a jumping off point for further/future exploration.

Conclusions

The process applied here provides great potential in becoming an actual price prediction application that can be used by the public. Although the complex manner in which the different features impact car pricing is constantly in flux, certain areas of importance were uncovered through exploration with eight distinct ML models. A 93.48% accuracy rating is substantial theoretically, though differences between actual versus predicted prices due to error with this model could amount to hundreds or even thousands of dollars. For buyers in the market, that would defeat the purpose of the

prediction model. Ideally, a near 100% accuracy rating should be sought after to achieve a true "working" model.

Machine learning appears to have the potential to change relationships between producers and consumers in positive ways. In this model, we have explored the potential for it to assist consumers in navigating the car market. A highly accurate machine learning price prediction model could potentially revolutionize the used automotive industry by giving buyers the power to locate the most reasonable deal. Consequently, sellers, both private party and dealer, will have to adjust their pricing schemes in order to accurately represent true market values. This may attract an even higher percentage of buyers to the car market.

In order to continue building towards an actual deployable model, more car data from other sources should be obtained and incorporated into the ML model. These sources will have to be varied in order to comprehensively capture the car sales market and will include other major online car platforms, as well as dealerships in order to represent the portion of car sales that are not transacted online. Obtaining data on local private party sales will be inconsistent and difficult to locate. A much larger data set would allow for more robust price predictions by the ML model. Although 7 different features, both numerical and categorical, were applied here, potentially more features could be included due to the broad importance value range for many different variables when considering car value.

Additionally, scrutinizing certain aspects of the data cleaning and pre-processing could lead to more accurate methods of filling in missing values or scaling the data, which could provide a data set that the ML model optimally uses for predictive analysis. Immediate next steps include loading the current prediction model into a Heroku-hosted web application.

Technologies Used

- Jupyter Notebook
- Python - Pandas, Seaborn, Matplotlib, Numpy, scikit-learn, yellowbrick, XGBoost
- Bootstrap
- CSV file
- Machine Learning Models - Linear Regression, Ridge, Lasso, KNN, Random Forest Regressor, Bagging Regressor, Adaboost Regressor, XGBoost Regressor