Capstone 2 - US Used Car Sales Predictive Model

1. **Objective:**

To develop a predictive model that automates the domestic used car sales market appraisal and estimation/evaluation process. The main driving force behind the development of this model is to minimize laborious efforts that go into researching used car sales data (e.g. what can I get for a 2011 toyota camry SE?) by deploying a model built using historical data with the ability to consume new data (leveraging Bayesian principles – ideal, future state model).

1. **Assumptions and Exclusions:**

*B1. Key Assumptions/Comments 🡪*

* Prices are set by a free market (consumer demand versus available supply)
* Some potential key features were missing from this analysis: color, general condition of the car (ordinal scaling), damage/vehicle inspection history, etc.
* Accounted for missing values and erroneous/extreme data points after visualizing distributions of each variable

*B2. Key Exclusions* 🡪

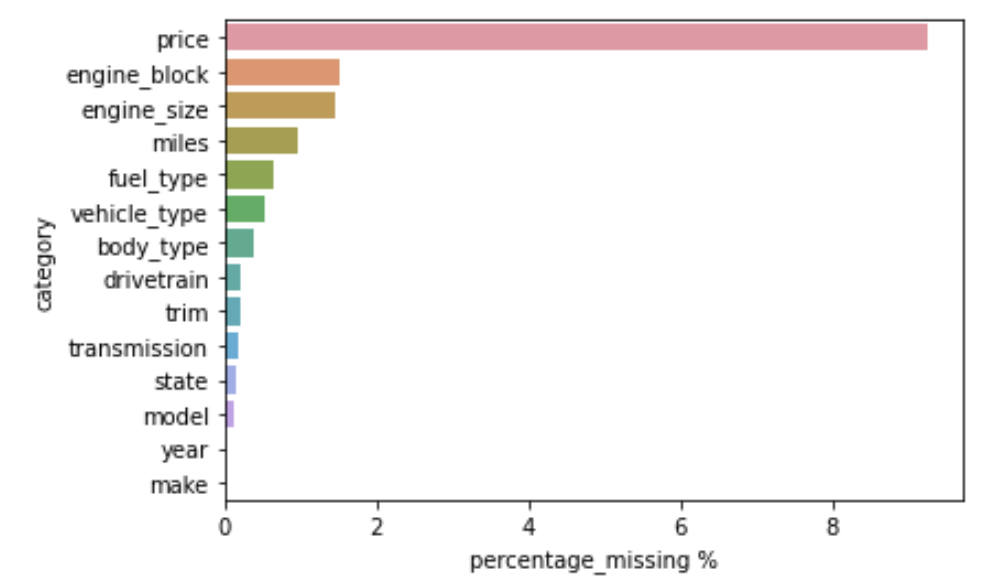
* Price data was missing from about 9.24% of the dataset; these observations were dropped as a result
* Some features in the original dataset were excluded due to them having an assumed low impact: 'id', 'vin', 'stock\_no', 'street', 'seller\_name', 'zip', 'city'
  + The vin and stock number are too granular for this analysis and would result in a ridiculously high dimensionality
  + I left the ‘state’ categorical variable to incorporate the spatial context; this helps to better understand the geographical impact on price
  + The ‘seller\_name’ was not considered fundamental for this analysis, as I’m looking to find the true demand for a given car irrespective of what a specific entity sells said car for

1. **Data Wrangling, EDA and Preprocessing:**

*C1. Data Collection and Wrangling 🡪*

* Imported the original US Car sales dataset (from Kaggle.com)
  + 1) the original dataset contained 7,104,304 used car sales transactions and incorporated 21 features (see fig 1 on the next page)
    - Excluded 7 features as mentioned in section B2
  + 2) manually added in the manufacturer origin (country)

Table

Description automatically generated 

**Fig. 1** – Original Dataset Features **Fig. 2** – Missing data

*C2. Initial Inspection of numerical distributions 🡪*

A picture containing histogram

Description automatically generated Chart, histogram

Description automatically generated

A picture containing text

Description automatically generated

**Figs. 3 and 4** – Price Distribution (left) and Mileage Distribution (right)

Chart, histogram

Description automatically generated Chart, histogram

Description automatically generated

**Fig. 5** – Year of Manufacturer Distribution **Fig. 6** – Engine Size Distribution

Chart

Description automatically generated

**Fig. 6** – Numerical Feature Correlation

*C3. Manual inspection of the values for categorical data and cleanup 🡪*

* Quite a bit of manual effort went into standardizing the naming conventions for the makes, models, fuel type and trim (example shown below)

Text, letter

Description automatically generated

**Fig. 7** – Model cleanup

Chart, box and whisker chart

Description automatically generated

**Fig. 8** – Top models sold by decade produced

Text

Description automatically generated

**Fig. 9** – Trim cleanup

*C4. Additional EDA 🡪*

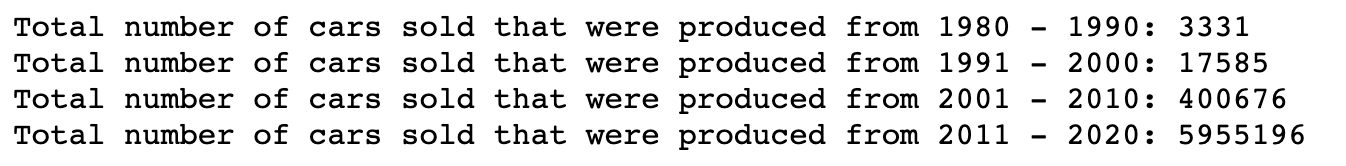
* Because there were quite a few categorical features, it was important to visualize their distribution/impact across the dataset in a variety of ways; mainly, how did the trend changed over time
* As one would have expected, the newer cars (>2015) yielded a higher sale price, but what was interesting is how the FWD was continuously the lowest while the RWD/4WD median price changed based on the manufactured year
* Notice how the lowest valued cars appear to be made from 2000-2005; why could this be? Is it purely a supply v demand (meaning, most cars older then 15 years are kept in pristine condition/low mileage and are coveted by car enthusiasts) or is there a theme related to how a car is manufactured (do people just generally like the look/feel of cars in the 80s/90s moreso than the early 2000s)?
  + I believe the prevailing phenomena is the former of what was outlined above. There are significantly less cars sold that were produced before 2000 (see **Fig 12** on the next page)
  + Hence, there were 1787x more cars sold that were produced from 2011-2020 when compared to ones produced from 1980-1990

Chart, histogram

Description automatically generated Chart, line chart, histogram

Description automatically generated

**Fig. 10** – Median Price per Drivetrain Type by Year **Fig. 11** – Median Price per Transmission Type by Year



**Fig. 12** - # of cars sold per manufactured decade

* For the next investigated step, I cleaned up the fuel types, and consolidated the naming conventions
* After the cleanup, the distribution of counts/observations of the various fuel types can be seen in **Fig. 13,** shownbelow
* Additionally, I wanted to visualize the relationship between price, fuel type and year of manufacture to get a better understanding of how the market has changed over time

Chart, bar chart

Description automatically generated Chart, line chart, histogram

Description automatically generated

**Fig. 13** – Fuel Type Counts  **Fig. 14** – Price based on Fuel Type and Year

* As part of the analysis, I included the “country of origin” feature to better visualize how price is affected by the manufacturer’s country; as one would expect, Italy, Germany and England are consistently higher than the other nations
* Older foreign cars appear to be more valued/coveted by car enthusiasts

Chart, line chart, histogram

Description automatically generated

**Fig. 15** – Country of Origin, Price and Year of Manufacture

Chart, scatter chart

Description automatically generated

**Fig. 16** – Price v Miles by Make

* The scatter plot shown above is a bit busy, but it uses a log scale for the x axis (price)
* The emphasis on this plot is to identify some outliers, as mileage and price are correlated
  + Notice how some cars > $100k while sporting >150,000 miles; this is quite compelling and shows that some models have been deemed to warrant a heavy premium compared to the population as a whole
* The second scatter plot, shown below, brings the data up a few levels
  + Here you can see that Italian models are typically higher in price and lower in mileage when compared to the other countries of origin
  + Swedish cars are typically the cheapest and have quite a bit of mileage

Chart, scatter chart

Description automatically generated

**Fig. 17** – Price v Miles by Country of Origin

Chart, line chart

Description automatically generated

**Fig. 18** – Closer look at Italian makes

Chart, line chart, histogram

Description automatically generated

**Fig. 19** – Closer look at German makes

*C5. Preprocessing 🡪*

* After dropping the rows that had no associated price, I had quite a few variables with missing values (see **Fig. 20** below)
* Initially, I planned on using the missingforests imputation algorithm (or something a bit more intelligent), but it required too much time and was computationally expensive
  + Realistically, because the % of values missing is so small, a simple imputation approach is sufficient
* Because my intention was to use a tree-based regressor or a neural network, it made sense to just fill the missing categorical values with “missing” and the missing numerical values with something that would identify them as being different: miles = -10,000, year = 1900, engine\_size = 1.0

Table

Description automatically generated

**Fig. 20** – remaining missing values

Text

Description automatically generated with medium confidence

**Fig. 21** – imputing the rest of the missing values

* After completing the imputation of all missing values, I went ahead and created a custom dummy transformer that would handle the high dimensionality encoding requirements (fig 22 on the next page); this was necessary for testing out some of the regressors (RandomForestRegressor, LinearRegressor, XGBRegressor)
* Once the categorical variables were successfully encoded, I was looking to scale the numerical variables
* The aformentioned steps were part of the pre\_pipeline, that when combined with the model, would complete the full\_pipeline as shown in Fig. 23 on the next page

A picture containing text

Description automatically generated

**Fig. 22** – Custom Dummy Encoder

Graphical user interface, text, email

Description automatically generated

**Fig. 23** – Initial Pipeline

1. **Modelling:**

*D1. Initial Discovery Phase 🡪*

* Based on preprocessing steps defined in section C5, I was able to test 4 different algorithms on the training data (80/20 split)
  + The caveat here is that because my categorical variables created a wide encoded dataset, I was forced to exclude the trim and model features to run some of the more complex models
  + This was not ideal and led me to focusing my efforts on utilizing the LGBMRegressor, as it has more capabilities for handling higher dimensions with ease
  + The RidgeRegressor failed due to a lack of memory available
* The MSE scores of the initial models were as follows:

|  |  |  |
| --- | --- | --- |
| **Algorithm:** | **Timing to run model:** | **MSE:** |
| XGBRegressor | 2.97 hours | 28,930,331.50 |
| LinearRegressor | .131 hours | 91,910,606 |
| LGBMRegressor | .043 hours | 33,920,875.33 |
| RandomForestRegressor | .86 hours | 195,518,841 |

* Based on the above table, it appears as though the XBGRegressor could be the winner; after considering the necessary computational requirements and model flexibility, the LGBMRegressor was a more applicable algorithm
  + The main reason for going with the LGBMRegressor over the XGBRegressor was the native capabilities of the LGBMRegressor to effortlessly handle categorical variables without having to encode; this allowed me to incorporate all pertinent features instead of having to drop the model and trim features because of resource limitations

*D2. Baseline Model: LGBM Regressor (not optimized) 🡪*

Text

Description automatically generated

Chart, scatter chart

Description automatically generated

**Fig. 24** – Training Data Performance

* Ideally, I would’ve like to use the RSMLE (log error) metric, but the y\_pred forecasted some negative values and the RSMLE metric cannot handle negative values
  + Presumably because of the imputation method
  + Now, there are some simple solutions to get around the error, but the combination of MSE, MAE and sq\_rt(MSE) was sufficient for tuning

Chart, bar chart

Description automatically generated

**Fig. 25** – Training Model Feature Importance

* Good thing I kept the model and trim categories, as those appear to be the most important after fitting and evaluating the model on the training data; intuitively, this makes sense based on the EDA
  + Why? Because as I discussed in the EDA portion of the analysis, you have some high-mileage cars that yield incredibly high prices based on their “model” and type of model/ “trim”

*D3. Optimized Model: LGBM Regressor (tuning the hyperparameters) 🡪*

* To tune the base model, I went ahead and leveraged the sklearn RandomizedSearchCV package
* The hyperparameters in focus are shown below in **Fig.** 26

Graphical user interface, text, application, email

Description automatically generated

**Fig. 26** – RandomizedSearchCV Hyperparameter tuning

Text

Description automatically generated

**Fig. 27** – Timing, Best Parameters and Scoring of the Tuned Model

Chart

Description automatically generated

**Fig. 28** – Updated Feature Importance

* The base model had a SQRT MSE of 5,824.16 while the tuned model was 3,298.03; that marks a significant improvement
* The only thing left to do is to apply the tuned model to the test dataset after scaling the test dataset

Text

Description automatically generated with medium confidence Chart, scatter chart

Description automatically generated

**Fig. 29** – Test Data Scores and Visualizing 1000 Predictions

* The test dataset scored a little bit lower than the training data, but it is clear that the model is generic enough to be applied/utilized on unseen data

1. **Lessons Learned and Comments:**

* This project served as a great introduction into predictive modelling via leveraging regression techniques/algorithms
* Potential future improvements include the following:
  + This model can be improved upon by incorporating a bit more detail/features as described in the introductory section
  + With the addition of Bayesian principles, the model can serve as a foundation for creating a dynamic model that can learn and adapt as new information is digested
  + Eventually, one could fully automate the quoting process for buying used cars based on questionnaire and in-person validation of the questionnaire
    - Think of a program that requires user input (information about the car) and spits out how much a business should pay for the car in addition to how much they could get it they resold it (assuming X,Y and Z standard improvements based on % return on invested capital into renovations)
  + Create an additional model that aims at creating a more fruitful investment thesis; hence, do we go after smaller margin, higher moving models such as ford f150s or do we go after higher-end, slower-moving models that could take months to sell but lead to a higher margin per car sold
  + Potentially tie this in with Kelly blue book to further optimize a quoting program