Title: STAT 515 ~ Homework #6: Car Price Prediction Analysis

Done by: Sai Praneet Reddy Chinthala, Paani Narisetty

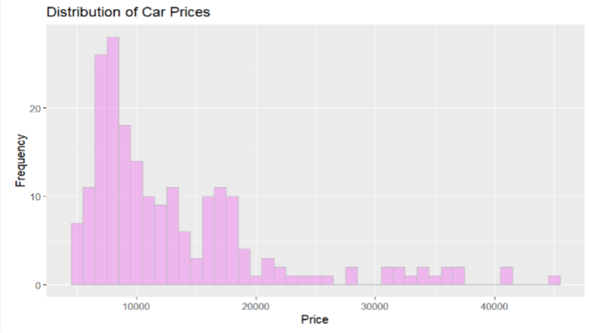
Date: 8th December 2024

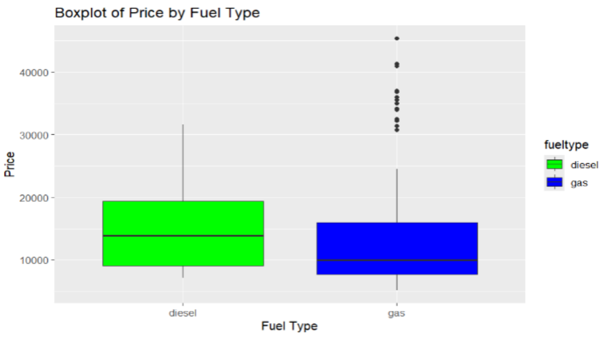
Introduction:

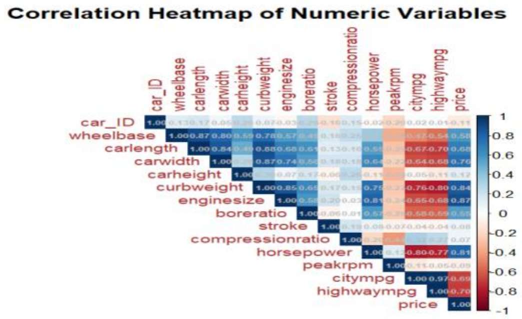
The given business problem deals with the creation of a model to predict the car prices for Geely Auto using the different data characteristics provided like car brand, fuel type, engine specifications, and car dimensions etc. The target variable is the car price, and the main aim is to find which variables significantly predict the price of a car and How well these variables explain car prices. To perform this project and obtain results successfully the required steps are to create the right model by exploring the dataset, performing data cleaning and preprocessing, and applying various statistical models, including linear regression and LASSO regression.

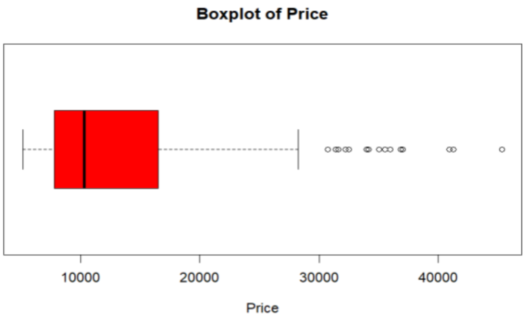
Data Preprocessing and visualization:

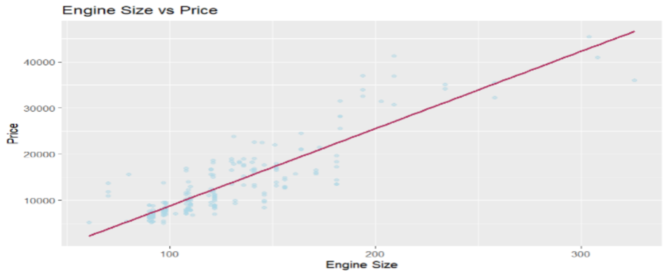
Loading the dataset and dictionary is the first step. There were no significant missing values found in the dataset. However, categorical variables like 'brand' had some inconsistencies (e.g., spelling errors like "toyouta" for "toyota"). These were corrected to ensure consistency. All categorical variables were converted to factors to perform the modelling process. A new feature, brand, was taken from the CarName variable by removing the brand name to allow a better analysis. A few visualizations are done to understand the data better and are shown below.


Distribution of Car Prices


Boxplot of Price by Fuel Type


Correlation Heatmap of Numeric Variables

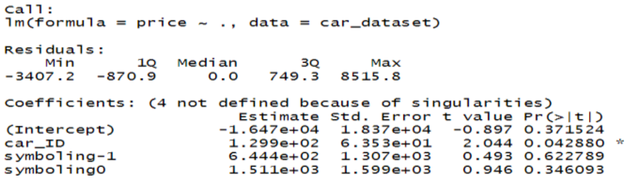

Boxplot of Price


Engine Size vs Price

Interpretation: The histogram of car prices mentions that most cars are below $20,000 with a few highly priced beyond $40,000. The boxplot by fuel type shows that gas-powered cars have a bigger price range when compared to diesel cars. The correlation heatmap shows that there are strong positive correlations between price and variables like engine size. The scatterplot of engine size versus price also shows a positive relationship, with bigger engine sizes generally indicate higher prices.

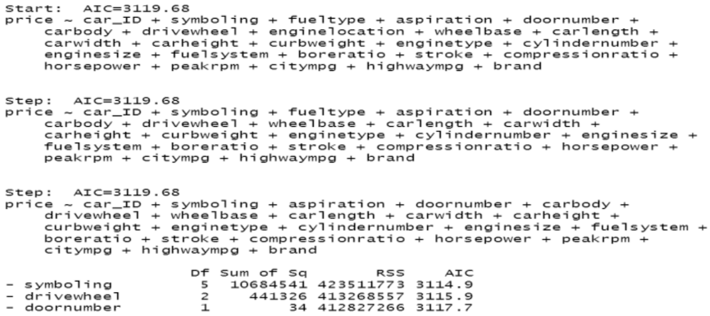
Model Development:

Firstly, linear regression with all predictors was used to create a model. Below is an extract from the summary of the linear regression model.



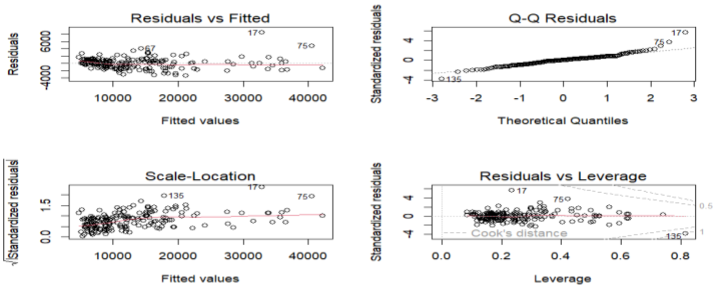
```
Call:
lm(formula = price ~ ., data = car_dataset)

Residuals:
    Min      1Q  Median      3Q     Max 
-3407.2  -870.9     0.0   749.3  8515.8 

Coefficients: (4 not defined because of singularities)
                       Estimate Std. Error t value Pr(>|t|)  
(Intercept)          -1.647e+04  1.837e+04  -0.897 0.371524  
car_ID                1.299e+02  6.353e+01   2.044 0.042880 *
symboling-1           6.444e+02  1.307e+03   0.493 0.622789  
symboling0            1.511e+03  1.599e+03   0.946 0.346093  
```

Stepwise regression (both forward and backward) was applied using the AIC criterion to decrease the predictors present while maintaining performance of the model. This allowed the number of predictors to be reduced. Based on the summary extract below it can be seen that the stepwise regression model selected 23 out of 26 predictors by removing enginelocation, fuelsystem, and citympg, which gives the model with the lowest AIC of 3114.9.

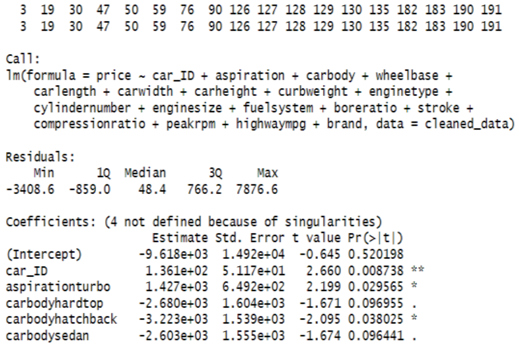


```
Start:  AIC=3119.68
price ~ car_ID + symboling + fueltype + aspiration + doornumber + 
    carbody + drivewheel + enginelocation + wheelbase + carlength + 
    carwidth + carheight + curbweight + enginetype + cylindernumber + 
    enginesize + fuelsystem + boreratio + stroke + compressionratio + 
    horsepower + peakrpm + citympg + highwaympg + brand


Step:  AIC=3119.68
price ~ car_ID + symboling + fueltype + aspiration + doornumber + 
    carbody + drivewheel + wheelbase + carlength + carwidth + 
    carheight + curbweight + enginetype + cylindernumber + enginesize + 
    fuelsystem + boreratio + stroke + compressionratio + horsepower + 
    peakrpm + citympg + highwaympg + brand


Step:  AIC=3119.68
price ~ car_ID + symboling + aspiration + doornumber + carbody + 
    drivewheel + wheelbase + carlength + carwidth + carheight + 
    curbweight + enginetype + cylindernumber + enginesize + fuelsystem + 
    boreratio + stroke + compressionratio + horsepower + peakrpm + 
    citympg + highwaympg + brand

                   Df Sum of Sq       RSS    AIC
- symboling         5  10684541 423511773 3114.9
- drivewheel        2    441326 413268557 3115.9
- doornumber        1        34 412827266 3117.7
```

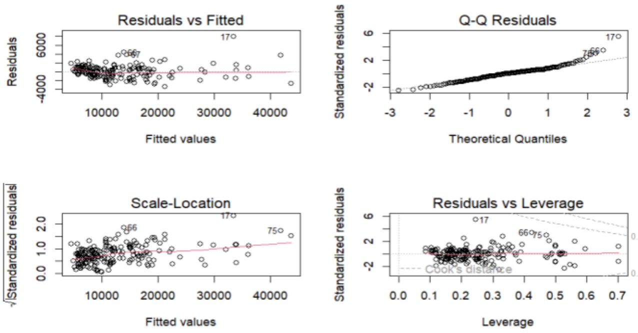
Now, VIFs were calculated to check the multicollinearity. Any predictor with a VIF above the threshold was removed from the model to ensure that the regression is not biased. Based on this, 4 residual plots are shown together. Both the Residuals vs. Fitted plot and the ScaleLocation plot show a funnel shape, indicating that the model's error variance increases with larger fitted values. The Q-Q plot has deviated away from the straight line, particularly in the tails, indicating that residuals may not be normally distributed. There are a couple of points that have high leverage but do not seem to be outliers with large residuals. It is possible that these mentioned issues may indicate the model that does not fit the data as well.



High-leverage points, which can affect the results, were identified using leverage values. These points were removed from the dataset to create a refined model. After removing high-leverage points and non-significant predictors, a refined model was developed. This model showed improved coefficients and better fit to the data. The summary shows that the residuals, ranging from -3408.6 to 7876.6, indicates that variability in the model's predictions decreased from before. Also, the refined plots are shown to further understand the model. From the Residuals vs. Fitted plot, the residuals lie within a fairly small area around the zero line, showing that the data reasonably has captured the basic trend in the data.



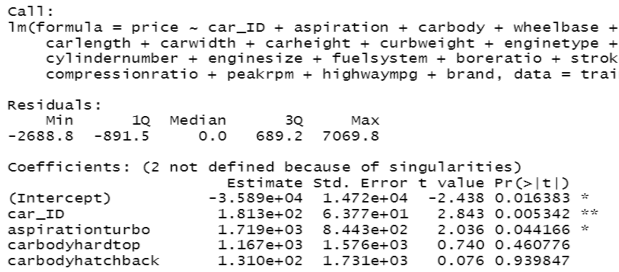
```
  3  19  30  47  50  59  76  90 126 127 128 129 130 135 182 183 190 191
  3  19  30  47  50  59  76  90 126 127 128 129 130 135 182 183 190 191

Call:
lm(formula = price ~ car_ID + aspiration + carbody + wheelbase +
    carlength + carwidth + carheight + curbweight + enginetype +
    cylindernumber + enginesize + fuelsystem + boreratio + stroke +
    compressionratio + peakrpm + highwaympg + brand, data = cleaned_data)

Residuals:
    Min      1Q  Median      3Q     Max
-3408.6  -859.0    48.4   766.2  7876.6

Coefficients: (4 not defined because of singularities)
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)      -9.618e+03  1.492e+04  -0.645 0.520198
car_ID            1.361e+02  5.117e+01   2.660 0.008738 **
aspirationturbo   1.427e+03  6.492e+02   2.199 0.029565 *
carbodyhardtop   -2.680e+03  1.604e+03  -1.671 0.096955 .
carbodyhatchback -3.223e+03  1.539e+03  -2.095 0.038025 *
carbodysedan     -2.603e+03  1.555e+03  -1.674 0.096441 .
```



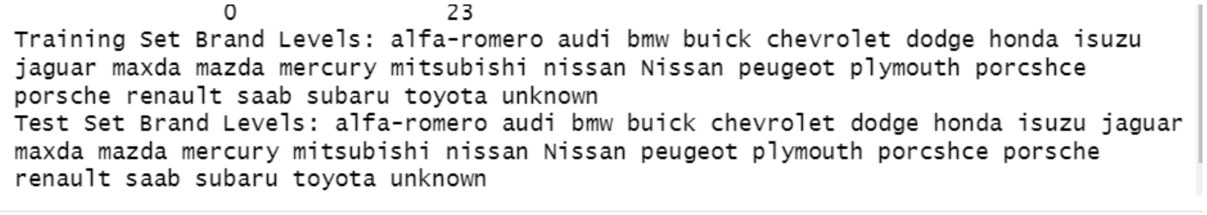
Model Evaluation:

The dataset is divided into two parts: a training subset and a test subset. The first 165 top rows go to the training and the following 40 to testing. The 'brand' variable, most likely representing car brand, is considered as a categorical factor in both sets to ensure the categories are dealt with appropriately. In the test set, any unexpected or wrong levels are changed to their correct value ("toyouta" to "toyota", "vokswagen" to "volkswagen" and "vw" to "volkswagen"). Any missing values in the 'brand' variable in the test set are replaced by the label "unknown". Unknown" is added to the training set if it is not already a level and the factor levels for 'brand' are set to be the same in both the training and test sets to ensure that both the sets have the same levels of the 'brand' factor. This way, the model will know how to handle this 'brand' variable for training and testing even when it gets unexpected or missing values.

A linear regression model, train_model is created on car prices based on a set of features in the training set. A function is created that checks for missing values and replaces NA values in the 'brand' column with "unknown." Then, the 'brand' variable is converted to a factor, with its levels aligned to those in the training set. This makes sure that, in case "unknown" is not present in the 'brand' factor of the training set, it gets included as a level. That will make the model robust while making predictions with unexpected or missing values. Attached is an extract of the summary of the model. From this it can be interpreted that Minimum is -2688.8 and the max is 7069.8. Also attached is the training dataset Brand levels.



```
Call:
lm(formula = price ~ car_ID + aspiration + carbody + wheelbase +
    carlength + carwidth + carheight + curbweight + enginetype +
    cylindernumber + enginesize + fuelsystem + boreratio + stroke +
    compressionratio + peakrpm + highwaympg + brand, data = train_set)

Residuals:
    Min      1Q  Median      3Q     Max
-2688.8  -891.5     0.0   689.2  7069.8

Coefficients: (2 not defined because of singularities)
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -3.589e+04  1.472e+04  -2.438 0.016383 *
car_ID              1.813e+02  6.377e+01   2.843 0.005342 **
aspirationturbo     1.719e+03  8.443e+02   2.036 0.044166 *
carbodyhardtop      1.167e+03  1.576e+03   0.740 0.460776
carbodyhatchback    1.310e+02  1.731e+03   0.076 0.939847
```



```
            0                23
Training Set Brand Levels: alfa-romero audi bmw buick chevrolet dodge honda isuzu
jaguar maxda mazda mercury mitsubishi nissan Nissan peugeot plymouth porcshce
porsche renault saab subaru toyota unknown
Test Set Brand Levels: alfa-romero audi bmw buick chevrolet dodge honda isuzu jaguar
maxda mazda mercury mitsubishi nissan Nissan peugeot plymouth porcshce porsche
renault saab subaru toyota unknown
```

Then, LASSO regression is fitted on the same data. The optimal regularization parameter (lambda) is determined by using cross-validation. This prevents overfitting of the model by penalizing its complexity. Using the LASSO model, the code predicts car prices on the test data and evaluates its performance on the RMSE metric. The smaller the RMSE, the better the predictive accuracy. In this case, the LASSO model has an RMSE of 3889.134 on the test set, which means that it can predict car prices with reasonable accuracy, though its practical significance depends on the specific context and the scale of the target variable.

Results:

The linear regression model showed a good fit to the data, but Lasso regression slightly improved the prediction accuracy by reducing overfitting. The Lasso model produced a slightly lower RMSE, indicating that it generalized better to unseen data.

- Key predictors of the car price are:
    - Engine size (enginesize)
    - Car dimensions (carwidth, carlength)
    - Weight (curbweight)
    - Fuel type and aspiration

Conclusion:

The findings suggest that Geely Auto should focus on the optimization of engine size, weight, and car dimensions to influence the pricing of the cars. Also, fuel type and aspiration are also important factors for pricing, which may indicate market trends towards certain car types, such as fuel-efficient cars.