Geely Auto – Price Prediction

Seema Sharanappa Kanaje
Multivariate Statistics Class

Abstract: *To predict significant variables of a car that affect price of a car for Geely Auto Dataset.*

Problem Statement:
A Chinese automobile company Geely Auto aspires to enter the US market by setting up their manufacturing unit there and producing cars locally to give competition to their US and European counterparts.

They have contracted an automobile consulting company to understand the factors on which the pricing of cars depends. Specifically, they want to understand the factors affecting the pricing of cars in the American market, since those may be very different from the Chinese market. The company wants to know:

- Which variables are significant in predicting the price of a car?
- How well those variables describe the price of a car.
- Based on various market surveys, the consulting firm has gathered a large dataset of different types of cars across the American market.

Business Goal:
We are required to model the price of cars with the available independent variables. It will be used by the management to understand how exactly the prices vary with the independent variables. They can accordingly manipulate the design of the cars, the business strategy etc. to meet certain price levels. Further, the model will be a good way for management to understand the pricing dynamics of a new market.

Executive Summary:

The main aim of this report is to build a model that predicts price of the cars only using important variables, how closely the variables are related to price and to help business how each variable is related with independent variables which helps in analysis and to formulate the business strategy. Before starting with analysis, a round of data cleaning was needed. Further, analysis of all numerical variable's vs price using pairwise plots. Box plots is used to do the analysis of all categorical variables. Few models were built by eliminating variables using Recursive Feature Extraction (RFE) and by observing R-Squared, R-Squared adj and P-value. Afterward, model is evaluated, and it provides a decisive conclusion on the significant variables in understanding the pricing dynamics of a new market

Selected Method:

- Import and Cleaning the data
- Segregating the columns into numerical variables and categorical variables
- Visualizing numerical variables using pairwise plots and categorical variables by using boxplots and dropping columns based on collinearity.
- One hot encoding was used for the categorical columns and changing all the string to numbers.
- Splitting data into train and test set.
- Rescaling the data using standard scaler.
- Fitted Regression model.
- Summary of the model.
- New Model was built using Recursive Feature Extraction.
- New models were built by dropping columns with high P value and by observing R-Squared and R-Squared adj
- Evaluated the model by predicting data on test data along with Root mean square, Error Terms and Residual plots
- Concluding on the significant variables

Import and Cleaning data:

```
jaguar         3
vw             2
maxda          2
renault        2
mercury        1
porcshce       1
toyouta        1
vokswagen      1
Nissan         1
Name: car_company, dtype: int64
```

Figure 1

In Figure 1, we can see that data was entered with different abbreviation and spellings errors. We take these values and replace with correct values. For instance, Volkswagen as been abbreviated as ‘vw’, misspelt as ‘Vokswagen’.

For analysis, we would only require name of car company but in the given data car company is mentioned along with the car model. We extract only company name and consider it as a independent variable for model building

Data Description:

Table 1 gives details of data dictionary along with type of the variable

Variable	Description	Variable Type
Symboling	that the auto is risky, -3 that it is probably prety safe.	Numerical
wheelbase	Wheelbase of car.	Numerical
carheight	Height of car.	Numerical
curbweight	The weight of a car without occupants or baggage.	Numerical
enginesize	Size of car.	Numerical
boreratio	Boreratio of car.	Numerical
stroke	Stroke or volume	Numerical
compressionratio	Compression ratio of car.	Numerical
horsepower	Horsepower	Numerical
peakrpm	Car peak rpm.	Numerical
price	Price of car.	Numerical
'fueltype'	Car fuel type i.e gas or diesel.	Categorical
'aspiration'	Aspiration used in a car.	Categorical
'doornumber'	Number of doors in a car.	Categorical
'carbody'	Body of car	Categorical
'drivewheel'	Type of drive wheel.	Categorical
'enginelocation'	Location of car engine.	Categorical
'enginetype'	Type of engine	Categorical
'cylindernumber'	Cylinder placed in the car.	Categorical
'fuelsystem'	Fuel system of car.	Categorical
car_company	Car brand name	Categorical

Table 1

Data Visualization:

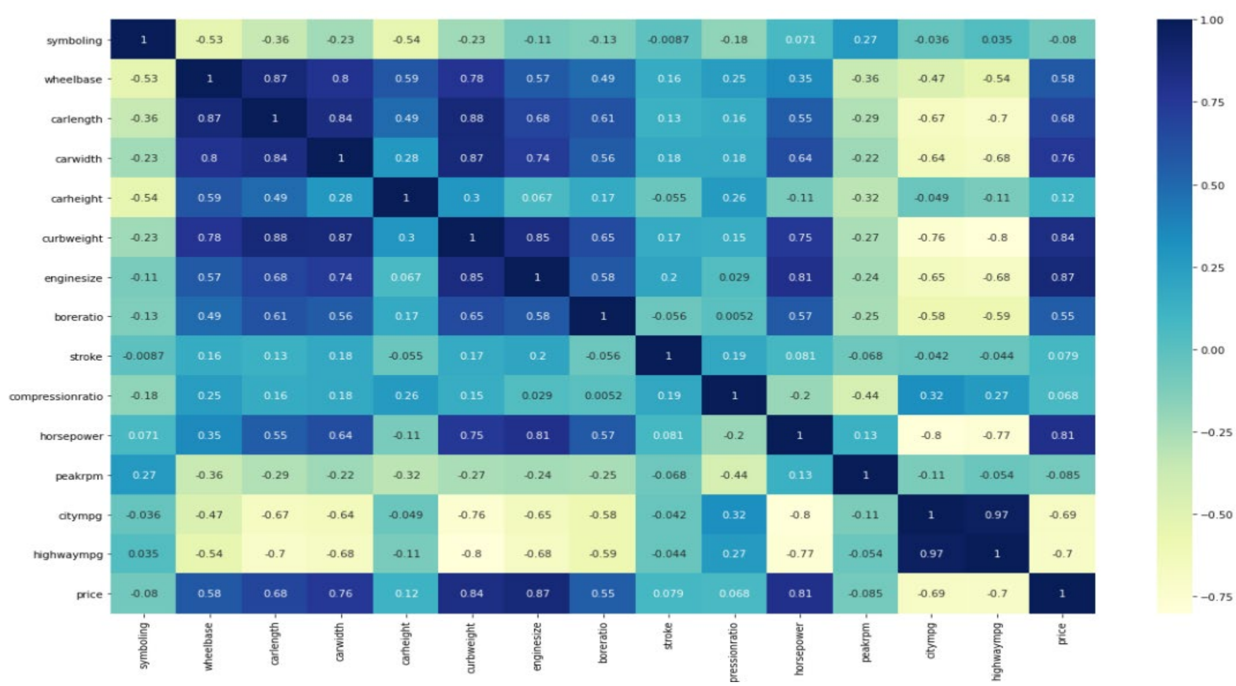


Figure 2

From the above heatmap(Figure 2), we see the correlation of Price with other variables:

- price is negatively correlated with symboling, peakrpm, citympg and highwaympg.
- price has a very low correlation with carheight, stroke and compressionratio.
- price shows a decent correlation with wheelbase, carlength, boreratio.
- price is highly correlated to carwidth, crubweight, enginesize and horsepower

We also observe the following from Figure 2:

- carlength is highly correlated with carwidth. (corr = 0.84)
- carlength is highly correlated with wheelbase. (corr = 0.87)
- carwidth is highly correlated with crubweight. (corr = 0.87)
- crubweight is highly correlated with horsepower. (corr = 0.75)
- horsepower is highly correlated with enginesize. (corr = 0.81)
- highwaympg is highly correlated with citympg. (corr = 0.97)

To reduce multicollinearity, going to drop car length, car width as they both high correlation with curb weight. Therefore, we keep curb weight which is highly correlated to price and dr op car length and car width. Citympg and highwaympg have very low correlation with price. Therefore, dropping cithympg and highwaympg. We keep the rest of the variables as they ha ve high correlation.

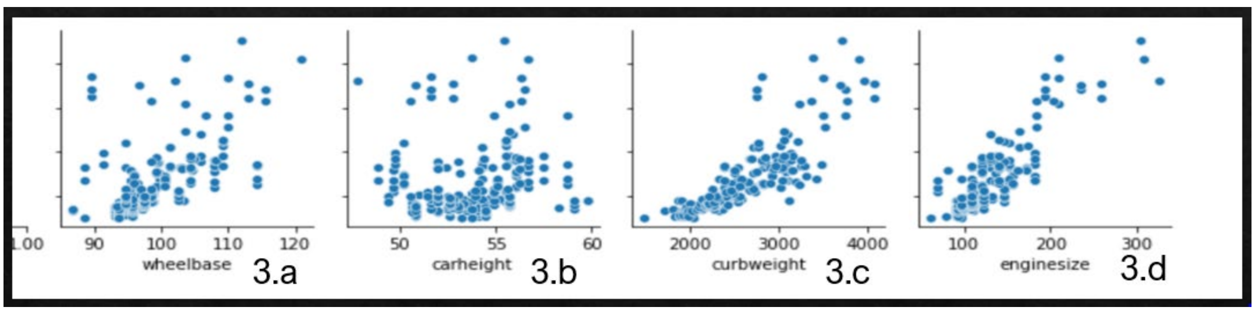


Figure 3

We are doing pairwise plot of numerical columns along with price to get more insight of the data (Referring Figure 3 and 4):

- Wheelbase is linearly correlated with price. (3.a)
- Carheight is slightly linearly correlated. (3.b)
- Curbweight is linearly correlated with price. (3.c)
- enginesize is linearly correlated with price. (3.d)
- boreratio is linearly correlated with price. (4.a)
- stroke is very negligible correlated and has has formed clustered at the bottom. (4.b)
- Compressionratio is segregated into two clusters. (4.c)
- Horsepower is linearly correlated with price. (4.d)
- Peakrpm is distributed. (4.e)

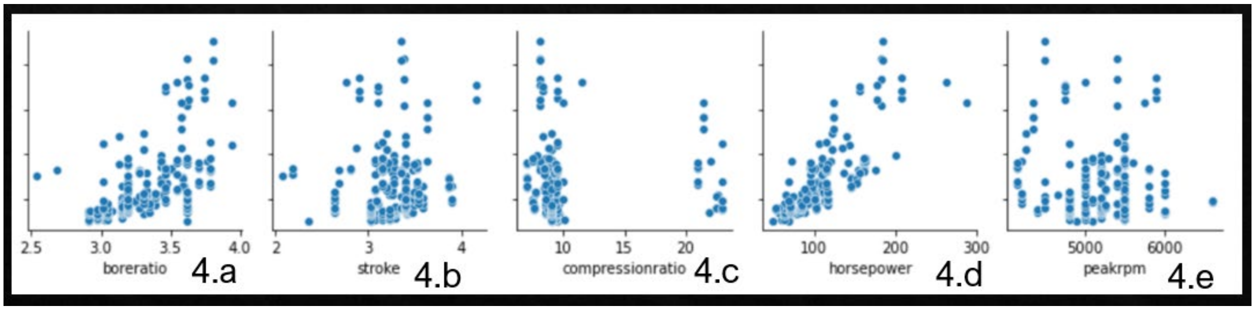


Figure 4

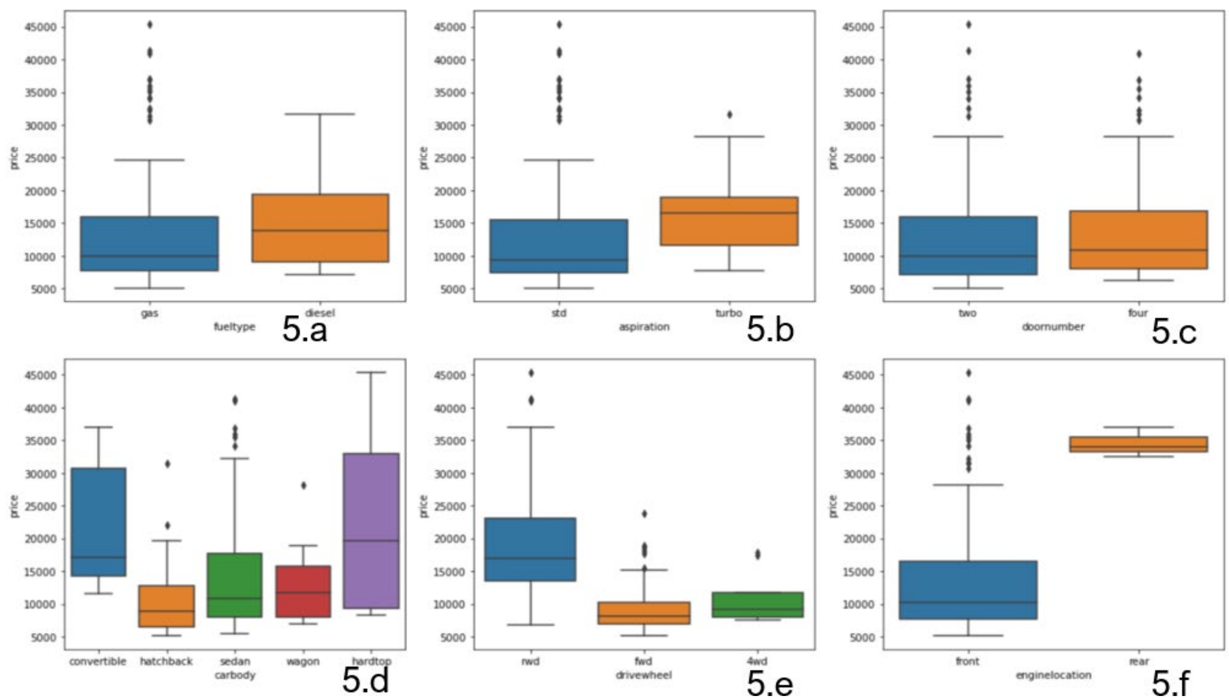


Figure 5

Plotting boxplots for categorical columns vs price(Referring Figure 5):

- Vehicles with Diesel as their fueltype is more expensive that gas. There are lot of outliers in the gas vehicles
- There is also increase in the price if aspiration is of type turbo. There are lot of outliers in the std type
- Number of doors in the car does not show much effect on the price.
- Hatchback,Sedan and Wagon are less expensive than the hardtop and convertible. Hardtop being expensive out of all. There are more than few outliers in sedan.
- Drivewheel with 'rwd' gets pricier than 'fwd' and '4wd'.
- Engine locations matter a lot for the price of the car, we can see that engine at the rear are almost more 50percent expensive than the engine at the front

With engine type 'ohcv', the car of the price gets expensive. ‘ohc’ and ‘ohcf’ are the cheaper engines among all. Others are moderately pricey.

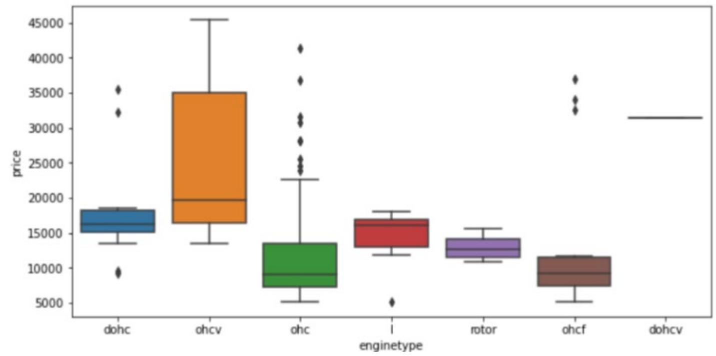


Figure 6

The price of the car also depends on the brand, in the below boxplot we can observe that 'BMW','BUICK','JAGUAR' and 'PORSCHE' are the expensive among all.Chervolet is least priced

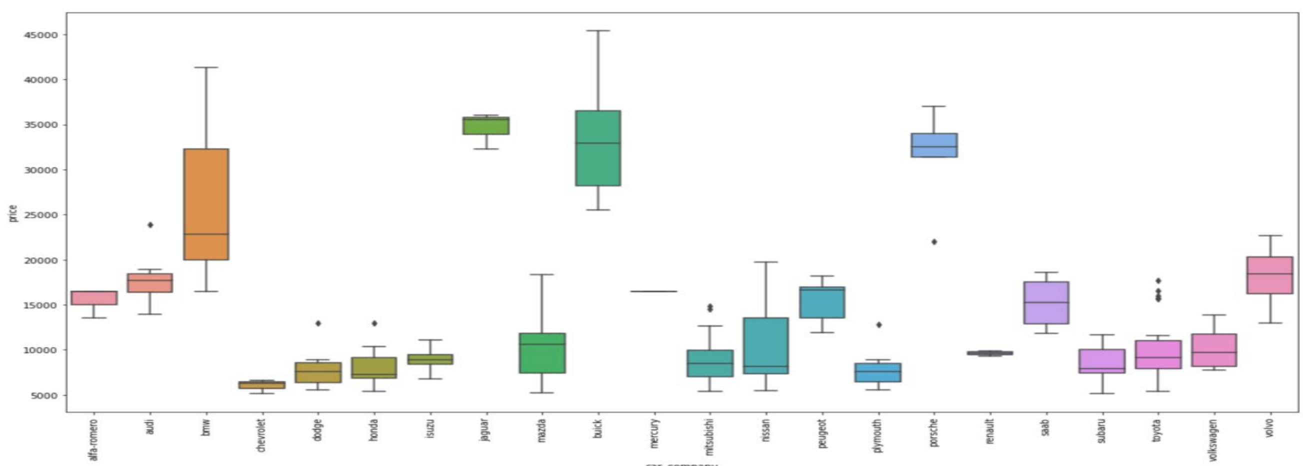


Figure 7

Data Preparation:

As the categorical columns have string values, we convert them into integers by using one hot encoding in python. We use get_dummies object from pandas to get integer values in dataframe which is equivalent to one hot encoding.

I have split the dataset into two parts 70 percent as training data set and another 30 percent as test data set.

Also, there are few variables like peakrpm, price, enginesize etc have large numbers and few more have extremely small values which needs to be rescaled. So that all variables have comparable scale otherwise we end up obtaining model with very large or very small co-efficient in comparison with other co-efficient. I have done scaling by using scikit learn library in python.

It can be done either by Minmax or Standard Scaler techniques. I have used Standard Scaler technique for scaling my numerical variables.

Model Selection Criteria:

To get the detailed statistics, linear regression model was built using statsmodel. As we can observe in the below summary that they are plenty of variable which is not viable. Also, we can observe that there are lot of variables (marked in red) have high p-values. Though R-Squared and R-Squared adj are big enough but still this model is not recommended.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.971
Model:                            OLS   Adj. R-squared:                  0.954
Method:                 Least Squares   F-statistic:                     56.84
Date:                Mon, 14 Dec 2020   Prob (F-statistic):           9.79e-51
Time:                        18:16:58   Log-Likelihood:                 50.986
No. Observations:                 143   AIC:                             6.027
Df Residuals:                      89   BIC:                             166.0
Df Model:                          53
Covariance Type:            nonrobust
==========================================================================================
                             coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------
const                     -1.4135      0.675     -2.094      0.039      -2.755      -0.072
symboling                  0.0054      0.042      0.127      0.899      -0.079       0.090
wheelbase                  0.2429      0.068      3.583      0.001       0.108       0.378
carheight                 -0.1951      0.053     -3.665      0.000      -0.301      -0.089
curbweight                 0.2512      0.115      2.179      0.032       0.022       0.480
enginesize                 1.6652      0.325      5.120      0.000       1.019       2.311
boreratio                 -0.7216      0.154     -4.681      0.000      -1.028      -0.415
stroke                    -0.1518      0.062     -2.447      0.016      -0.275      -0.029
compressionratio          -0.1071      0.284     -0.377      0.707      -0.672       0.458
horsepower                -0.0774      0.163     -0.475      0.636      -0.401       0.246
peakrpm                    0.1632      0.046      3.513      0.001       0.071       0.255
fueltype_gas              -0.8807      0.518     -1.699      0.093      -1.911       0.150
aspiration_turbo           0.4425      0.133      3.336      0.001       0.179       0.706
doornumber_two            -0.0831      0.072     -1.156      0.251      -0.226       0.060
carbody_hardtop           -0.5840      0.261     -2.241      0.028      -1.102      -0.066
carbody_hatchback         -0.5762      0.201     -2.864      0.005      -0.976      -0.176
carbody_sedan             -0.5303      0.205     -2.588      0.011      -0.937      -0.123
carbody_wagon             -0.4841      0.216     -2.237      0.028      -0.914      -0.054
drivewheel_fwd             0.0157      0.116      0.135      0.893      -0.215       0.247
drivewheel_rwd             0.0710      0.166      0.428      0.670      -0.259       0.401
enginelocation_rear        0.5721      0.302      1.894      0.062      -0.028       1.172
enginetype_dohcv           1.5851      0.789      2.010      0.047       0.018       3.152
enginetype_l               1.3551      0.360      3.767      0.000       0.640       2.070
enginetype_ohc             0.0713      0.206      0.347      0.730      -0.338       0.480
enginetype_ohcf            0.8099      0.195      4.150      0.000       0.422       1.198
enginetype_ohcv            0.0328      0.177      0.186      0.853      -0.318       0.384
enginetype_rotor           2.1890      0.536      4.081      0.000       1.123       3.255
cylindernumber_five        1.7003      0.648      2.623      0.010       0.412       2.988
cylindernumber_four        2.6142      0.840      3.112      0.002       0.945       4.283
```

Figure 8

Therefore, I am using RFE to automatically pick important features/variables. RFE is a Recursive Feature Elimination, to select features by recursively considering smaller and smaller sets of features. RFE is performed by scikit learn library and feature selection module. I did select 20 features as my n_feature parameter to RFE module. Below are the variables that are chosen by the RFE model (Figure 9):

```
Index(['curbweight', 'enginesize', 'carbody_hardtop', 'carbody_hatchback',
       'carbody_sedan', 'carbody_wagon', 'enginelocation_rear',
       'enginetype_dohcv', 'enginetype_l', 'enginetype_rotor',
       'cylindernumber_three', 'cylindernumber_twelve', 'cylindernumber_two',
       'car_company_audi', 'car_company_bmw', 'car_company_buick',
       'car_company_peugeot', 'car_company_porsche', 'car_company_saab',
       'car_company_volvo'],
      dtype='object')
```

Figure 9

These are the variables rejected by the RFE.

```
Index(['symboling', 'wheelbase', 'carheight', 'boreratio', 'stroke',
       'compressionratio', 'horsepower', 'peakrpm', 'fueltype_gas',
       'aspiration_turbo', 'doornumber_two', 'drivewheel_fwd',
       'drivewheel_rwd', 'enginetype_ohc', 'enginetype_ohcf',
       'enginetype_ohcv', 'cylindernumber_five', 'cylindernumber_four',
       'cylindernumber_six', 'fuelsystem_2bbl', 'fuelsystem_4bbl',
       'fuelsystem_idi', 'fuelsystem_mfi', 'fuelsystem_mpfi',
       'fuelsystem_spdi', 'fuelsystem_spfi', 'car_company_chevrolet',
       'car_company_dodge', 'car_company_honda', 'car_company_isuzu',
       'car_company_jaguar', 'car_company_mazda', 'car_company_mercury',
       'car_company_mitsubishi', 'car_company_nissan', 'car_company_plymouth',
       'car_company_renault', 'car_company_subaru', 'car_company_toyota',
       'car_company_volkswagen'],
      dtype='object')
```

Figure 10

Figure 11, model was built using the features provide bye RFE. But still, we can see there are variables with high P-values

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.931
Model:                            OLS   Adj. R-squared:                  0.921
Method:                 Least Squares   F-statistic:                     93.26
Date:                Mon, 14 Dec 2020   Prob (F-statistic):           4.53e-63
Time:                        18:17:34   Log-Likelihood:                -11.520
No. Observations:                 143   AIC:                             61.04
Df Residuals:                     124   BIC:                             117.3
Df Model:                          18
Covariance Type:            nonrobust
========================================================================================
                           coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------
const                    0.1929      0.147      1.312      0.192      -0.098       0.484
curbweight               0.4409      0.088      5.006      0.000       0.267       0.615
enginesize               0.3268      0.087      3.738      0.000       0.154       0.500
carbody_hardtop         -0.4344      0.246     -1.764      0.080      -0.922       0.053
carbody_hatchback       -0.3883      0.152     -2.558      0.012      -0.689      -0.088
carbody_sedan           -0.3195      0.149     -2.141      0.034      -0.615      -0.024
carbody_wagon           -0.4859      0.166     -2.924      0.004      -0.815      -0.157
enginelocation_rear      1.2635      0.456      2.770      0.006       0.361       2.166
enginetype_dohcv         0.3121      0.401      0.779      0.438      -0.481       1.105
enginetype_l             0.1297      0.104      1.247      0.215      -0.076       0.336
enginetype_rotor         0.3574      0.091      3.907      0.000       0.176       0.538
cylindernumber_three     0.4158      0.202      2.060      0.041       0.016       0.815
cylindernumber_twelve    0.3206      0.367      0.874      0.384      -0.406       1.047
cylindernumber_two       0.3574      0.091      3.907      0.000       0.176       0.538
car_company_audi         0.4959      0.141      3.512      0.001       0.216       0.775
car_company_bmw          1.1969      0.130      9.206      0.000       0.940       1.454
car_company_buick        0.8290      0.157      5.281      0.000       0.518       1.140
car_company_peugeot     -0.2861      0.139     -2.058      0.042      -0.561      -0.011
car_company_porsche      0.9451      0.287      3.294      0.001       0.377       1.513
car_company_saab         0.2985      0.176      1.693      0.093      -0.050       0.647
car_company_volvo        0.3523      0.134      2.626      0.010       0.087       0.618
==============================================================================
Omnibus:                       20.275   Durbin-Watson:                   2.008
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               53.594
Skew:                           0.501   Prob(JB):                     2.30e-12
Kurtosis:                       5.827   Cond. No.                     5.31e+16
==============================================================================
```

Figure 11

Figure12, though there is no change in R-Squared and R-Squared adj values. But still, we can observe that p value of 'cylindernumber_twelve' is greater than 0.05. We eliminate that and rebuild the model

OLS Regression Results

Dep. Variable:	price	R-squared:	0.931
Model:	OLS	Adj. R-squared:	0.921
Method:	Least Squares	F-statistic:	99.02
Date:	Mon, 14 Dec 2020	Prob (F-statistic):	6.04e-64
Time:	18:18:16	Log-Likelihood:	-11.868
No. Observations:	143	AIC:	59.74
Df Residuals:	125	BIC:	113.1
Df Model:	17		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1923	0.147	1.310	0.193	-0.098	0.483
curbweight	0.4397	0.088	5.001	0.000	0.266	0.614
enginesize	0.3308	0.087	3.797	0.000	0.158	0.503
carbody_hardtop	-0.4338	0.246	-1.764	0.080	-0.920	0.053
carbody_hatchback	-0.3871	0.152	-2.554	0.012	-0.687	-0.087
carbody_sedan	-0.3185	0.149	-2.137	0.035	-0.613	-0.024
carbody_wagon	-0.4838	0.166	-2.916	0.004	-0.812	-0.155
enginelocation_rear	1.1057	0.408	2.710	0.008	0.298	1.913
enginetype_l	0.1306	0.104	1.258	0.211	-0.075	0.336
enginetype_rotor	0.3596	0.091	3.938	0.000	0.179	0.540
cylindernumber_three	0.4182	0.201	2.076	0.040	0.019	0.817
cylindernumber_twelve	0.3037	0.366	0.830	0.408	-0.420	1.028
cylindernumber_two	0.3596	0.091	3.938	0.000	0.179	0.540
car_company_audi	0.4953	0.141	3.513	0.001	0.216	0.774
car_company_bmw	1.1931	0.130	9.198	0.000	0.936	1.450
car_company_buick	0.8235	0.157	5.260	0.000	0.514	1.133
car_company_peugeot	-0.2875	0.139	-2.071	0.040	-0.562	-0.013
car_company_porsche	1.0967	0.210	5.210	0.000	0.680	1.513
car_company_saab	0.2989	0.176	1.698	0.092	-0.049	0.647
car_company_volvo	0.3509	0.134	2.620	0.010	0.086	0.616

Omnibus:	19.423	Durbin-Watson:	2.020
Prob(Omnibus):	0.000	Jarque-Bera (JB):	50.132
Skew:	0.482	Prob(JB):	1.30e-11
Kurtosis:	5.736	Cond. No.	5.13e+16

Figure 12

Now there is slight increase in R-squared adj. As P value of 'enginetype_l' is greater than 0.05, we will rebuild the model and check.

OLS Regression Results

Dep. Variable:	price	R-squared:	0.930
Model:	OLS	Adj. R-squared:	0.922
Method:	Least Squares	F-statistic:	105.4
Date:	Mon, 14 Dec 2020	Prob (F-statistic):	8.12e-65
Time:	18:18:38	Log-Likelihood:	-12.262
No. Observations:	143	AIC:	58.52
Df Residuals:	126	BIC:	108.9
Df Model:	16		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1826	0.146	1.249	0.214	-0.107	0.472
curbweight	0.4097	0.080	5.118	0.000	0.251	0.568
enginesize	0.3717	0.072	5.174	0.000	0.230	0.514
carbody_hardtop	-0.4315	0.246	-1.757	0.081	-0.917	0.054
carbody_hatchback	-0.3803	0.151	-2.516	0.013	-0.679	-0.081
carbody_sedan	-0.3078	0.148	-2.076	0.040	-0.601	-0.014
carbody_wagon	-0.4607	0.163	-2.821	0.006	-0.784	-0.137
enginelocation_rear	1.0757	0.406	2.650	0.009	0.272	1.879
enginetype_l	0.1401	0.103	1.359	0.176	-0.064	0.344
enginetype_rotor	0.3837	0.086	4.439	0.000	0.213	0.555
cylindernumber_three	0.4174	0.201	2.075	0.040	0.019	0.816
cylindernumber_two	0.3837	0.086	4.439	0.000	0.213	0.555
car_company_audi	0.5039	0.140	3.588	0.000	0.226	0.782
car_company_bmw	1.1710	0.127	9.235	0.000	0.920	1.422
car_company_buick	0.8074	0.155	5.204	0.000	0.500	1.114
car_company_peugeot	-0.2773	0.138	-2.008	0.047	-0.551	-0.004
car_company_porsche	1.0769	0.209	5.156	0.000	0.664	1.490
car_company_saab	0.3150	0.175	1.803	0.074	-0.031	0.661
car_company_volvo	0.3557	0.134	2.662	0.009	0.091	0.620

Omnibus:	16.922	Durbin-Watson:	2.016
Prob(Omnibus):	0.000	Jarque-Bera (JB):	42.459
Skew:	0.404	Prob(JB):	6.03e-10
Kurtosis:	5.544	Cond. No.	7.00e+16

Figure 13

After eliminating ‘enginetype_l’, model has still slightly higher P-value for ‘Car_Peugeot’

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.930
Model:                            OLS   Adj. R-squared:                  0.922
Method:                 Least Squares   F-statistic:                     105.4
Date:                Mon, 14 Dec 2020   Prob (F-statistic):           8.12e-65
Time:                        18:20:16   Log-Likelihood:                -12.262
No. Observations:                 143   AIC:                             58.52
Df Residuals:                     126   BIC:                             108.9
Df Model:                          16
Covariance Type:            nonrobust
========================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------
const                   0.1826      0.146      1.249      0.214      -0.107       0.472
curbweight              0.4097      0.080      5.118      0.000       0.251       0.568
enginesize              0.3717      0.072      5.174      0.000       0.230       0.514
carbody_hardtop        -0.4315      0.246     -1.757      0.081      -0.917       0.054
carbody_hatchback      -0.3803      0.151     -2.516      0.013      -0.679      -0.081
carbody_sedan          -0.3078      0.148     -2.076      0.040      -0.601      -0.014
carbody_wagon          -0.4607      0.163     -2.821      0.006      -0.784      -0.137
enginelocation_rear     1.0757      0.406      2.650      0.009       0.272       1.879
enginetype_rotor        0.3837      0.086      4.439      0.000       0.213       0.555
cylindernumber_three    0.5575      0.288      1.933      0.055      -0.013       1.128
cylindernumber_two      0.3837      0.086      4.439      0.000       0.213       0.555
car_company_audi        0.5039      0.140      3.588      0.000       0.226       0.782
car_company_bmw         1.1710      0.127      9.235      0.000       0.920       1.422
car_company_buick       0.8074      0.155      5.204      0.000       0.500       1.114
car_company_peugeot    -0.1371      0.138     -0.997      0.321      -0.409       0.135
car_company_porsche     1.0769      0.209      5.156      0.000       0.664       1.490
car_company_saab        0.3150      0.175      1.803      0.074      -0.031       0.661
car_company_volvo       0.3557      0.134      2.662      0.009       0.091       0.620
==============================================================================
Omnibus:                       16.922   Durbin-Watson:                   2.016
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               42.459
Skew:                           0.404   Prob(JB):                     6.03e-10
Kurtosis:                       5.544   Cond. No.                     6.58e+16
==============================================================================
```

Figure 14

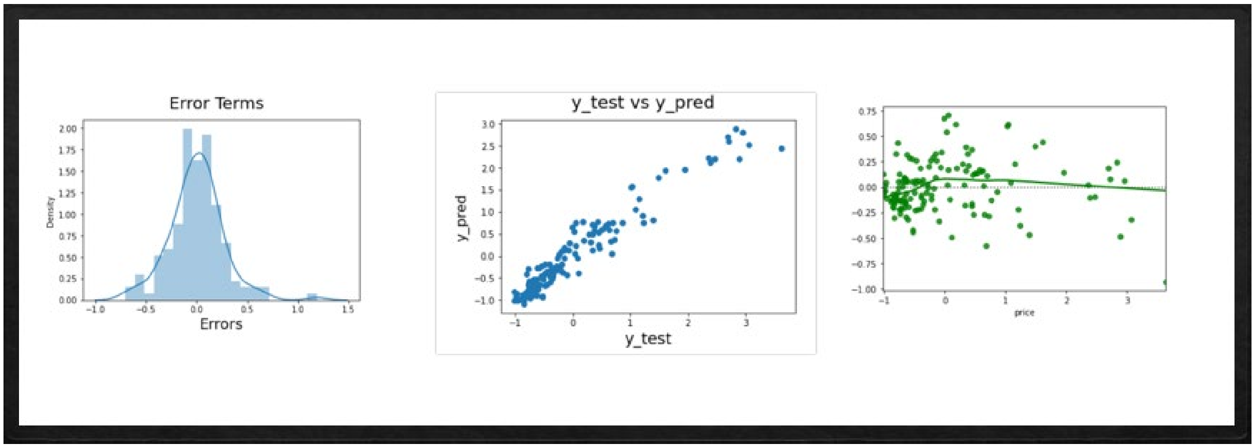
After eliminating ‘car_company_peugeot’ from the model. The model looks stable and all the variables are in limits. Also, there is not much change in the R-Squared value and R-Squared adj values. We go ahead and predict the price of the car with model.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.930
Model:                            OLS   Adj. R-squared:                  0.922
Method:                 Least Squares   F-statistic:                     112.4
Date:                Mon, 14 Dec 2020   Prob (F-statistic):           1.23e-65
Time:                        18:21:51   Log-Likelihood:                -12.824
No. Observations:                 143   AIC:                             57.65
Df Residuals:                     127   BIC:                             105.1
Df Model:                          15
Covariance Type:            nonrobust
========================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------
const                   0.1666      0.145      1.147      0.254      -0.121       0.454
curbweight              0.3598      0.062      5.762      0.000       0.236       0.483
enginesize              0.4097      0.061      6.728      0.000       0.289       0.530
carbody_hardtop        -0.4303      0.246     -1.752      0.082      -0.916       0.056
carbody_hatchback      -0.3783      0.151     -2.503      0.014      -0.677      -0.079
carbody_sedan          -0.3118      0.148     -2.103      0.037      -0.605      -0.018
carbody_wagon          -0.4482      0.163     -2.752      0.007      -0.771      -0.126
enginelocation_rear     1.0316      0.404      2.557      0.012       0.233       1.830
enginetype_rotor        0.4092      0.083      4.957      0.000       0.246       0.573
cylindernumber_three    0.5343      0.287      1.859      0.065      -0.034       1.103
cylindernumber_two      0.4092      0.083      4.957      0.000       0.246       0.573
car_company_audi        0.5441      0.135      4.044      0.000       0.278       0.810
car_company_bmw         1.1889      0.126      9.472      0.000       0.941       1.437
car_company_buick       0.8564      0.147      5.820      0.000       0.565       1.148
car_company_porsche     1.0916      0.208      5.239      0.000       0.679       1.504
car_company_saab        0.3565      0.170      2.101      0.038       0.021       0.692
car_company_volvo       0.4011      0.126      3.193      0.002       0.153       0.650
==============================================================================
Omnibus:                       18.094   Durbin-Watson:                   2.023
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               45.061
Skew:                           0.450   Prob(JB):                     1.64e-10
Kurtosis:                       5.598   Cond. No.                     4.87e+16
==============================================================================
```

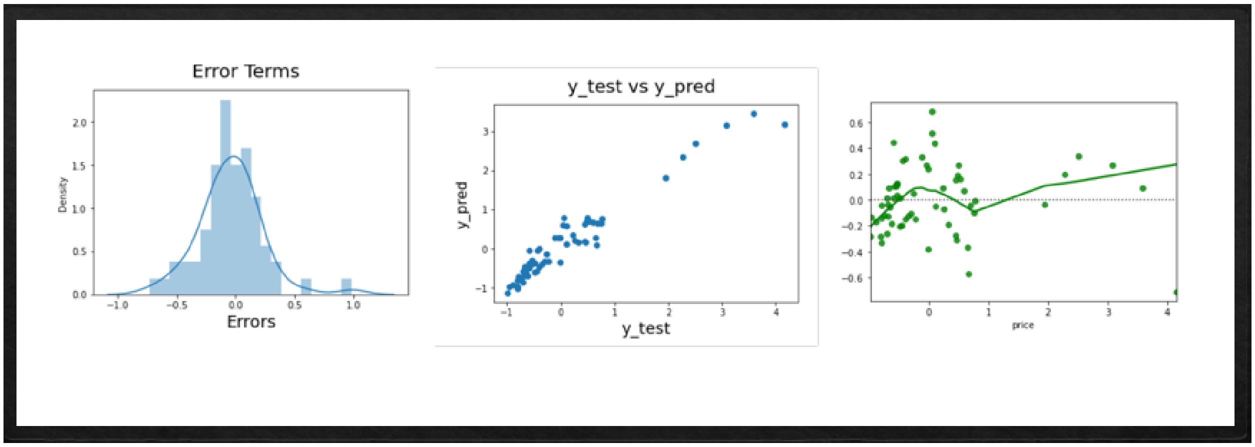
Figure 15

Model Comparison:

For training data,Plotting y_test and y_pred to understand the spread. From the spread we understand it is linear
An error term is a residual variable produced by a statistical or mathematical model, which is created when the model does not fully represent the actual relationship between the independent variables and the dependent variables. It is normally distributed which is good to go. Root mean square, value of zero is the perfect model. But there is deviation of 0.2646 which is acceptable. Lower RMS value better the model. Below graphs shows plot residuals, which are clustered on to left and rest are distributed. There is a spike as well in the below line where it is clustered which indicates there are outliers



Even in test data,there is deviation of 0.27146 which is acceptable.Plotting y_test and y_pred to understand the spread. From the spread we understand it is linear.Below graphs shows plot residuals, which are clustered on to left and rest are distributed. There is a spike as well in the below line where it is clustered which indicates there are outliers.



Conclusion: From our final model, the model looks stable and all the variables are in limits. Also, there is not much change in the R-Squared value and R-Squared adj values. Error terms is also normally distributed overall the model looks good. All the below variables would help in the prediction of the price.

1. Engine Size
2. Engine Location
3. Engine Type
4. Cylinder Number
5. Car Brand

References*:[1] Kaggle, geely-auto-car-price-linear-regression-assignment.*

[2] Towards Data Science, Predicting car price.

[3] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html