Department of Industrial and Management Engineering,
Indian Institute of Technology, Kanpur

MBA652A – Statistical Modelling for Business Analytics

Project 1: Car Price Prediction

Guided by:
DEVLINA CHATTERJEE
PhD (IISc Bangalore)

Submitted by:
Group 03:
ASHISH UBANA (18114005)
SAURABH GUPTA (19114010)
VIVEK PRAJAPAT (19114019)
PRADNESH LACHAKE (19114007)

INDEX

1. **INTRODUCTION**
   - 1.1. **Objective……4**
   - 1.2. **Methodology……..……..……4**
   - 1.3. **Economic Theory……..4**
2. **DATA**
   - 2.1. **Data Source……...4**
   - 2.2. **Variables……....5**
3. **DESCRIPTIVE ANALYSIS**
   - 3.1. **Statistical summary of data…….……6**
   - 3.2. **Correlation Matrix……...……7**
   - 3.3. **Variance Inflation Factor (VIF)……..……9**
   - 3.4. **Recursive Feature Elimination (RFE)……....……9**
4. **MODEL**
   - 4.1. **Linear Regression Model using Forward Selection……...……9**
   - 4.2. **Linear Regression Model using Backward Elimination…….…10**
   - 4.3. **Model Insights…….15**
5. **PYTHON CODE…….16**
6. **REFERENCES……...20**

ACKNOWLEDGEMENT:

We are highly indebted to Prof. Devlina Chatterjee, for her guidance and continuous support in completing this project. It is because of the knowledge and skills acquired during the course work, along with her comprehensive style of teaching, that we are able to understand the subject in a better way and are able to complete this modelling project successfully.

1. INTRODUCTION

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. For the same, they want to understand the factors on which the pricing of car depends. Basically, they want to identify those differentiating parameters which are different for American and Chinese market, so that they can make their strategy accordingly.

1.1 Objective

The objective of our project is as follows:

- Which variables are significant in predicting the price of a car.
- How well those variables describe the price of a car.

1.2 Methodology

The methodology for this project is as follows:

- Summary of data and data visualization.
- Start building model using uni-variate Linear Regression model, and then using multi-variate Linear Regression model.
- Further using Recursive Feature Elimination (RFE) method we have selected our attributes for the building our model
- Then on the basis of threshold p-values and VIF(Variance Inflation Factor) we have done backward elimination and have eliminated insignificant variables

1.3 Economic theory

From the model management can predict how exactly the prices are varying with the given variables and accordingly they can manipulate the design of the cars and can form their business strategy. The model will also help management to evaluate the pricing dynamics of the market.

2. DATA

2.1 Data Source

The data for our project is taken from UCI Machine learning repositories: the weblink is given below:

https://archive.ics.uci.edu/ml/datasets/Automobile

2.2 Variables

The **dependent variable** is of numerical type is denoted by "**price**" in the data set which represents the car price.

There are 25 **independent variables** shown below

1. Car_ID: Unique id of each observation (Integer)
2. Symboling: Its assigned insurance risk rating, A value of +3 indicates that the auto is risky, -3 that it is probably pretty safe.(Categorical)
3. carCompany: Name of car company (Categorical)
4. fueltype: Car fuel type i.e gas or diesel (Categorical)
5. aspiration: Aspiration used in a car (Categorical)
6. doornumber: Number of doors in a car (Categorical)
7. carbody: body of car (Categorical)
8. drivewheel: type of drive wheel (Categorical)
9. enginelocation: Location of car engine (Categorical)
10. wheelbase: Weelbase of car (Numeric)
11. carlength: Length of car (Numeric)
12. carwidth: Width of car (Numeric)
13. carheight: height of car (Numeric)
14. curbweight: The weight of a car without occupants or baggage. (Numeric)
15. enginetype: Type of engine. (Categorical)
16. cylindernumber: cylinder placed in the car (Categorical)
17. enginesize: Size of car (Numeric)
18. fuelsystem: Fuel system of car (Categorical)
19. boreratio: Boreratio of car (Numeric)
20. stroke: Stroke or volume inside the engine (Numeric)
21. compressionratio: compression ratio of car (Numeric)
22. horsepower: Horsepower (Numeric)
23. peakrpm: car peak rpm (Numeric)
24. citympg: Mileage in city (Numeric)
25. highwaympg: Mileage on highway (Numeric)

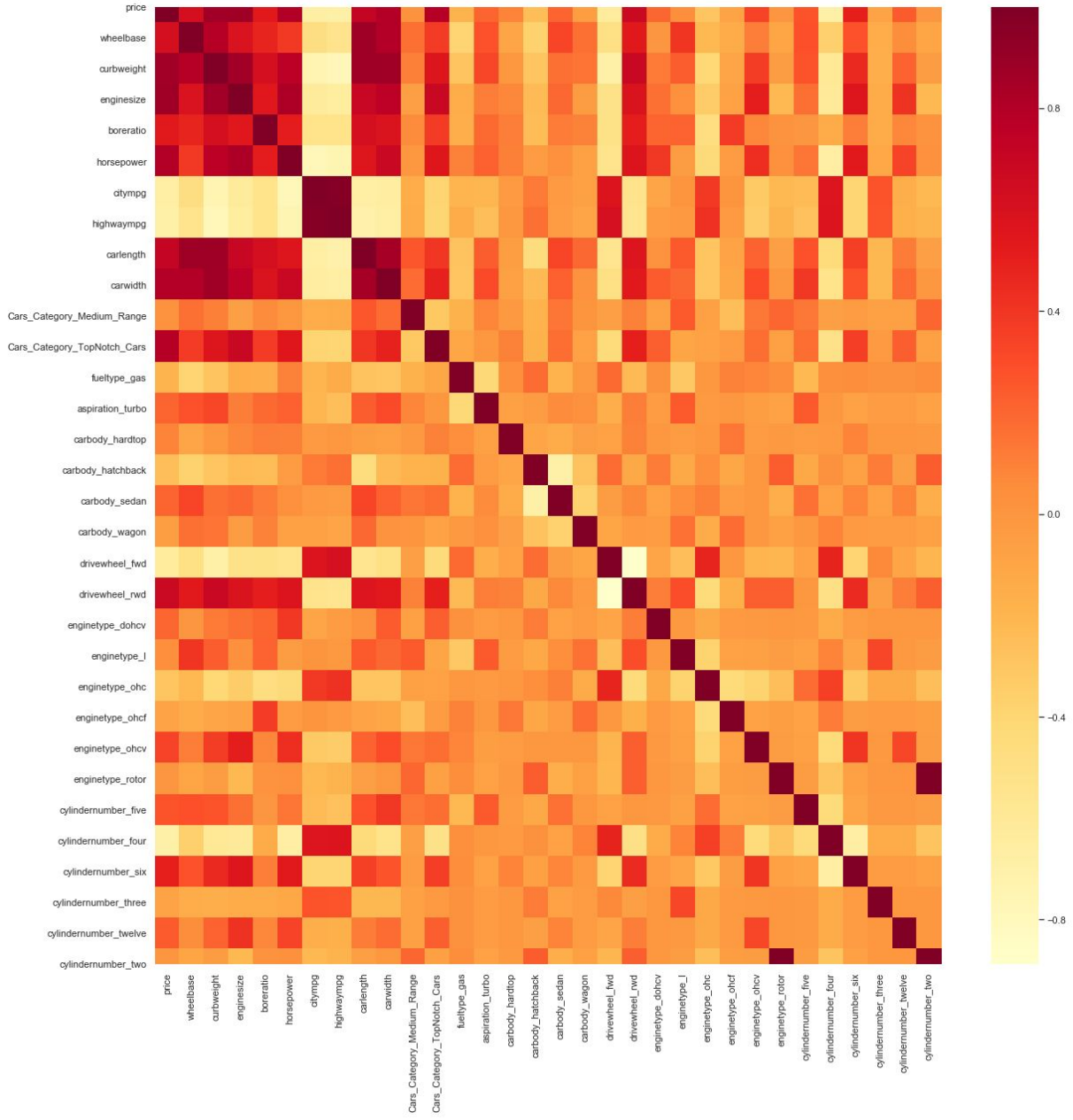
3. DESCRIPTIVE ANALYSIS

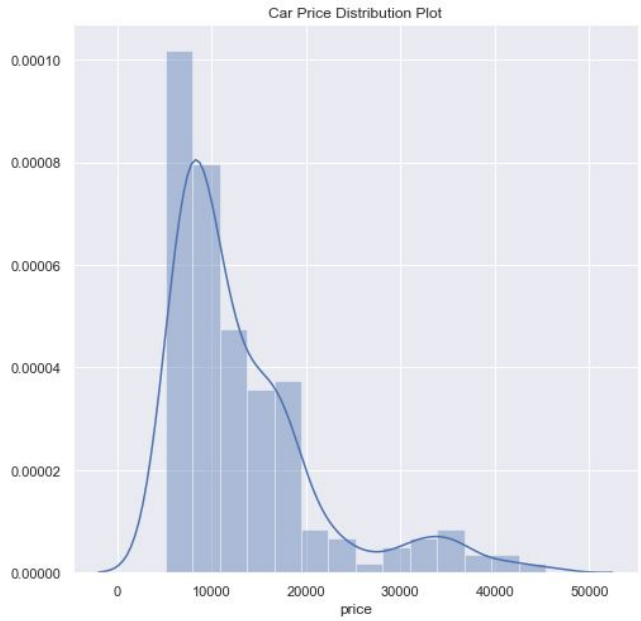
3.1 Statistical summary of data

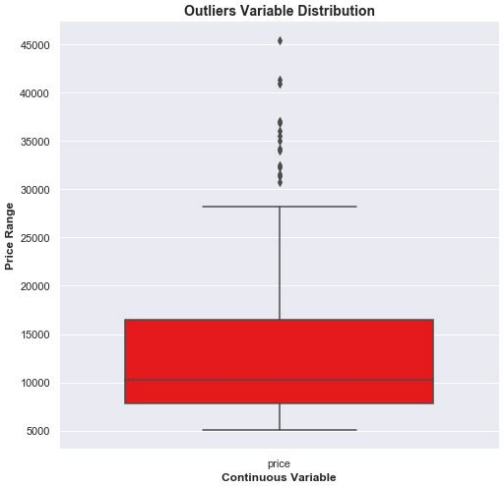
	wheelbase	carlength	carwidth	carheight	curbweight	enginesize	boreratio	stroke
count	205.0000	205.0000	205.0000	205.0000	205.0000	205.0000	205.0000	205.0000
mean	98.7566	174.0493	65.9078	53.7249	2555.5659	126.9073	3.3298	3.2554
std	6.0218	12.3373	2.1452	2.4435	520.6802	41.6427	0.2708	0.3136
min	86.6000	141.1000	60.3000	47.8000	1488.0000	61.0000	2.5400	2.0700
25%	94.5000	166.3000	64.1000	52.0000	2145.0000	97.0000	3.1500	3.1100
50%	97.0000	173.2000	65.5000	54.1000	2414.0000	120.0000	3.3100	3.2900
75%	102.4000	183.1000	66.9000	55.5000	2935.0000	141.0000	3.5800	3.4100
max	120.9000	208.1000	72.3000	59.8000	4066.0000	326.0000	3.9400	4.1700

	compressionratio	horsepower	peakrpm	citympg	highwaympg	price
count	205.0000	205.0000	205.0000	205.0000	205.0000	205.0000
mean	10.1425	104.1171	5125.1220	25.2195	30.7512	13276.7106
std	3.9720	39.5442	476.9856	6.5421	6.8864	7988.8523
min	7.0000	48.0000	4150.0000	13.0000	16.0000	5118.0000
25%	8.6000	70.0000	4800.0000	19.0000	25.0000	7788.0000
50%	9.0000	95.0000	5200.0000	24.0000	30.0000	10295.0000
75%	9.4000	116.0000	5500.0000	30.0000	34.0000	16503.0000
max	23.0000	288.0000	6600.0000	49.0000	54.0000	45400.0000

3.2 Correlation Matrix

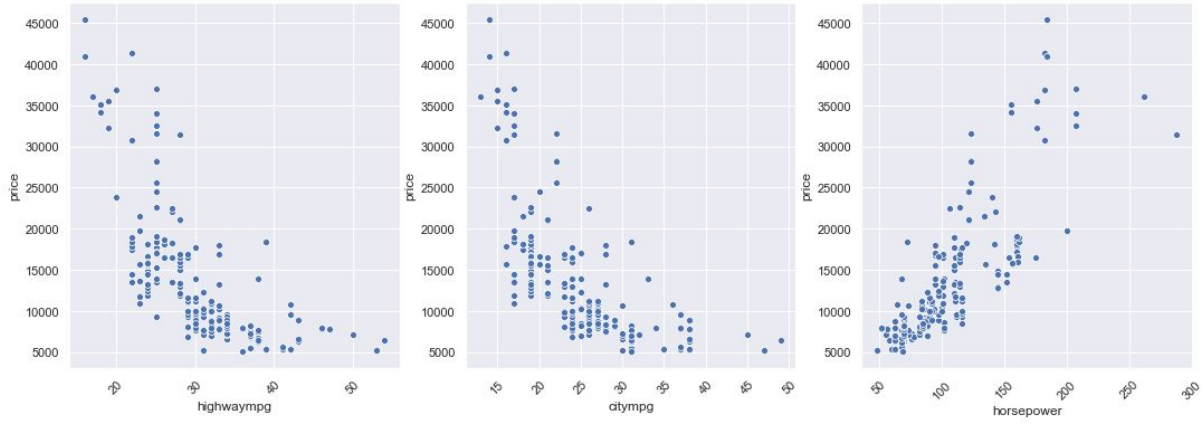


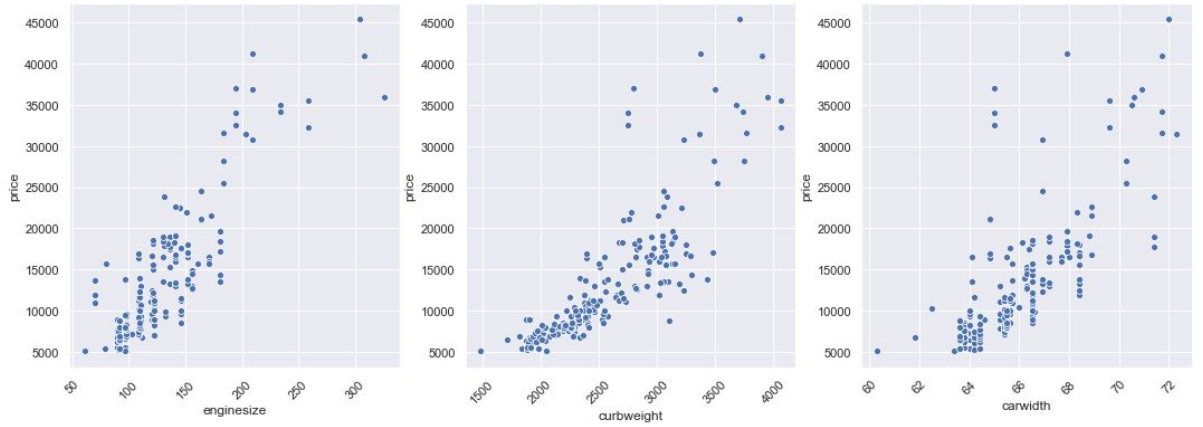




Car Price Distribution plot is positively skewed with mean 13276.71, standard deviation 7988.852332 and Q3 of 16503.00.

Scatter Plot





The scatter plot of the ‘horsepower’, ‘enginesize’, ‘curbweight’, ‘carwidth’ with shows a positive correlation with car price. Variables ‘highwaympg’, ‘citympg’ shows a negative correlation with the car price.

3.3 Variance Inflation Factor (VIF)

A variance inflation factor (VIF) provides a measure of multicollinearity among the independent variables in a multiple regression model. Detecting multicollinearity is important because while it does not reduce the explanatory power of the model, it does reduce the statistical significance of the independent variables. A large VIF on an independent variable indicates a highly collinear relationship to the other variables that should be considered or adjusted for in the structure of the model and selection of independent variables[1].

$$\text{VIF} = \frac{1}{1-R_i^2}$$

The general thumb rule that we have used to interpret VIF for model is as follows: [3]

- VIF = 1: not correlated.
- VIF = between 1 and 5 : moderately correlated.
- VIF = greater than 5: highly correlated.

3.4 Recursive Feature Elimination

Recursive Feature Elimination is basically a backward selection of the predictors. This technique begins by building a model on the entire set of predictors and computing an importance score for each predictor. The least important predictor(s) are then removed, the model is re-built, and importance scores are computed again. In practice, the analyst specifies the number of predictor subsets to evaluate as well as each subset’s size. Therefore, the subset size is a tuning parameter for RFE. The subset size that optimizes the performance criteria is used to select the predictors based on the importance rankings[2].

Threshold Criteria: In our model we will iteratively drop the independent variables whose p-value is greater than 0.05 (i.e. 95 percent of confidence). VIF value is greater than 5.

4 MODEL

4.1 Linear Regression Model Using Forward Selection

- Model 1: const 0.0000, curbweight (beta coefficients- 0.8112), R^2 value obtained is 0.645.

- Model 2: const 0.0000, horsepower (beta coefficient- 0.3581), curbweight (beta coefficient- 0.5898, R^2 value obtained is 0.782.
- Model 3: const 0.0000, horsepower (beta coefficient-0.2288), curbweight (beta coefficient-0.3938), enginesize (beta coefficient-0.3400) and R^2 to be 0.82.

4.2 Linear Regression Model using Backward Elimination

	Model - 1			Model - 2			Model - 3		
VARIABLE	**Beta Value**	**p Values**	**VIF Values**	**Beta Value**	**p Values**	**VIF Values**	**Beta Value**	**p Values**	**VIF Values**
const	**0.7074**	**0**		**0.7124**	**0**		**0.6736**	**0**	
curbweight	**0.247**	**0**	**9.06**	**0.248**	**0**	**9.06**	**0.2384**	**0**	**8.53**
horsepower	**0.215**	**0**	**5.61**	**0.2324**	**0**	**4.69**	**0.2273**	**0**	**4.53**
carwidth	**0.227**	**0**	**6.24**	**0.2328**	**0**	**6.15**	**0.2408**	**0**	**5.72**
Cars_Category_TopNotch_Cars	**1.061**	**0**	**2.17**	**1.0502**	**0**	**2.14**	**1.0599**	**0**	**2.13**
carbody_hardtop	**-0.2211**	**0.303**	**1.45**	**-0.2314**	**0.279**	**1.45**	**-0.25**	**0.238**	**1.28**
carbody_hatchback	**-0.6016**	**0**	**6.15**	**-0.6198**	**0**	**5.95**	**-0.6311**	**0**	**3.76**
carbody_sedan	**-0.4691**	**0.002**	**9.73**	**-0.4951**	**0.001**	**9.05**	**-0.5123**	**0.001**	**5.24**
carbody_wagon	**-0.5865**	**0**	**3.48**	**-0.6111**	**0**	**3.31**	**-0.6269**	**0**	**2.36**
enginetype_dohcv	**-1.0254**	**0.006**	**1.84**	**-0.9329**	**0.008**	**1.67**	**-0.8773**	**0.01**	**1.6**
enginetype_ohc	**0.126**	**0.061**	**5.97**	**0.1212**	**0.07**	**5.92**	**-0.1104**	**0.088**	**5.65**
enginetype_ohcv	**-0.1968**	**0.105**	**1.63**	**-0.2225**	**0.059**	**1.53**	**-0.218**	**0.063**	**1.52**
cylindernumber_five	**-0.3815**	**0.022**	**2.82**	**-0.3529**	**0.03**	**2.66**	**-0.2894**	**0.03**	**1.97**

cylindernumber_four	**-0.4642**	**0**	**15.92**	**-0.4483**	**0.001**	**15.27**	**-0.3896**	**0**	**8.99**
cylindernumber_six	**-0.1381**	**0.368**	**4.73**						
cylindernumber_twelve	**-0.296**	**0.311**	**1.66**	**-0.0857**	**0.480**	**3.7**			
R-squared			**0.936**			**0.936**			**0.936**
Adj R -squared			**0.929**			**0.929**			**0.929**

	Model - 4			**Model - 5**			**Model - 6**		
VARIABLE	**Beta Value**	**p Values**	**VIF Values**	**Beta Value**	**p Values**	**VIF Values**	**Beta Value**	**p Values**	**VIF Values**
const	**0.5631**	**0**		**0.5665**	**0**		**0.4821**	**0**	
curbweight	**0.2474**	**0**	**8.51**	**0.2288**	**0.001**	**8.25**	**0.246**	**0**	**8.1**
horsepower	**0.2179**	**0**	**4.17**	**0.2151**	**0**	**4.17**	**0.2289**	**0**	**4.13**
carwidth	**0.2386**	**0**	**5.65**	**0.2397**	**0**	**5.65**	**0.2058**	**0**	**5.08**
Cars_Category_TopNotch_ Cars	**1.0606**	**0**	**2.08**	**1.1121**	**0**	**1.83**	**1.1276**	**0**	**1.83**
carbody_hardtop									
carbody_hatchback	**-0.5214**	**0**	**3.31**	**-0.4954**	**0**	**3.1**	**-0.4798**	**0**	**2.63**
carbody_sedan	**-0.4054**	**0**	**4.64**	**-0.3789**	**0.001**	**4.33**	**-0.3696**	**0.001**	**3.52**

carbody_wagon	-0.5259	0	2.22	-0.5137	0	2.2	-0.5091	0	1.94
enginetype_dohcv	-0.8407	0.014	1.58	-0.883	0.01	1.57	-0.8214	0.016	1.54
enginetype_ohc	0.1029	0.11	5.6						
enginetype_ohcv	-0.1975	0.089	1.52	-0.2098	0.072	1.51	-0.1484	0.18	1.43
cylindernumber_five	-0.2793	0.036	1.93	-0.2024	0.104	1.63			
cylindernumber_four	-0.3811	0	8.09	-0.3351	0	6.8	-0.2608	0.001	5.76
cylindernumber_six									
cylindernumber_twelve									
R-squared			0.935			0.934			0.932
Adj R -squared			0.929			0.928			0.927

	Model - 7			Model - 8			Model - 9		
VARIABLE	Beta Value	p Values	VIF Values	Beta Value	p Values	VIF Values	Beta Value	p Values	VIF Values
const	0.4472	0.001		0.4709	0.001		0.244	0.038	
curbweight	0.2449	0	8.1						
horsepower	0.2146	0	3.81	0.3021	0	2.72	0.3599	0	2.44

carwidth	0.206	0	5.08	0.3523	0	2.22	0.3652	0	2.12
Cars_Category_ TopNotch_Cars	1.1456	0	1.81	1.2141	0	1.73	1.2895	0	1.7
carbody_hardtop									
carbody_hatchback	-0.4744	0	2.47	-0.5316	0	2.4	-0.4859	0	1.1
carbody_sedan	-0.3619	0.002	3.35	-0.3839	0.001	3.35	-0.3518	0.0032	1.22
carbody_wagon	-0.5038	0	1.89	-0.423	0.002	1.71	-0.4023	0.0045	1.02
enginetype_dohcv	-0.7421	0.028	1.49	-1.2424	0	1.24	-1.445	0	1.22
enginetype_ohc									
enginetype_ohcv									
cylindernumber_five									
cylindernumber_four	-0.2382	0.002	5.66	-0.2504	0.002	5.66			
cylindernumber_six									
cylindernumber_twelve									
R-squared			0.931			0.924			0.918
Adj R -squared			0.927			0.919			0.914

	Model - 10			Model - 11		
VARIABLE	**Beta Value**	**p Values**	**VIF Values**	**Beta Value**	**p Values**	**VIF Values**
const	**-0.0748**	**0.057**		**-0.0925**	**0.009**	
curbweight						
horsepower	**0.3837**	**0**	**2.31**	**0.3847**	**0**	**2.28**
carwidth	**0.339**	**0**	**2.08**	**0.3381**	**0**	**2.07**
Cars_Category_ TopNotch_Cars	**1.3063**	**0**	**1.46**	**1.3179**	**0**	**1.45**
carbody_hardtop						
carbody_hatchback	**-0.1738**	**0.004**	**1.1**	**-0.1565**	**0.006**	**1.1**
carbody_sedan	**-0.0792**	**0.315**	**1.02**			
carbody_wagon						
enginetype_dohcv	**-1.4904**	**0**	**1.22**	**-1.5033**	**0**	**1.22**
enginetype_ohc						
enginetype_ohcv						
cylindernumber_five						
cylindernumber_four						
cylindernumber_six						

cylindernumber_twelve						
R-squared			**0.913**			**0.912**
Adj R -squared			**0.909**			**0.909**

4.3 Model Insights

At Model 9 we have seven variables with adj R^2 value 0.914. In order to explain maximum variability using minimum variable. We decided to drop 'carbody_sedan' by preliminary data visualization, p-value and correlation value and checked if there is substantial drop in R^2 value. After dropping the variable 'carbody_wagon', there was no substantial drop in R^2 value (0.05). So, we further proceeded with dropping 'carbody_wagon'. It results in an increase in p-value of 'carbody_sedan' (p-value 0.315). The reason behind the increase of p value of this variable could be a strong negative correlation between 'carbody_sedan' and 'carbody_wagon'.

The R2 score of model 11 is 0.912. Hence, we can say that our model is good enough to predict the Car prices using predictor variables 'horsepower', 'carwidth', 'Cars_Category_TopNotch_Cars', 'carbody_hatchback', 'enginetype_dohcv'. The final multi-variate linear regression model is as follows:

Carprice = − 0.0925 + (0.3847×horsepower) + (0.3381×carwidth) + (1.3179×Carscategorytopnotchcars) − (0.1565×carbodyhatchback) − (1.5033×enginetypedohcv)

5. PYTHON CODE

```
# Importing Libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns


# Importing Libraries for building model

from sklearn import preprocessing

from sklearn.base import TransformerMixin

from sklearn.preprocessing import MinMaxScaler

import statsmodels.api as sm

from sklearn.feature_selection import RFE

from sklearn.linear_model import LinearRegression

from statsmodels.stats.outliers_influence import variance_inflation_factor


# Data Information

dataset.info()


# Data Description

dataset.describe()


# Outlier Detection

outliers = ['price']

sns.boxplot(data=dataset[outliers], orient="v", palette="Set1" ,whis=1.5,saturation=1, width=0.7)

plt.title("Outliers Variable Distribution")
```

```
plt.ylabel("Price Range")

plt.xlabel("Continuous Variable")

dataset.shape


# Histogram of price
plt.title('Car Price Distribution Plot')

sns.distplot(dataset['price'])


# Heat-Map
sns.heatmap(dataset_train.corr(), cmap="YlOrRd")
plt.show()

# Scatter Plot of independent variables vs Price
col = ['highwaympg','citympg','horsepower','enginesize','curbweight','carwidth']

fig,axes = plt.subplots(2,3)
for seg,col in enumerate(col):
    x,y = seg//3,seg%3
    an=sns.scatterplot(x=col, y='price' ,data=dataset, ax=axes[x,y])
    plt.setp(an.get_xticklabels(), rotation=45)
plt.subplots_adjust(hspace=0.5)
```

 # Linear Regression Model using Forward Selection

```
y = dataset_train.pop('price')
X = dataset_train
X_1 = X['curbweight']

# Add a constant
X_1c = sm.add_constant(X_1)

# Model1
lr_1 = sm.OLS(y, X_1c).fit()

# Finding beta coefficient and intercept
lr_1.params

# Summary
print(lr_1.summary())

X_2 = X[['horsepower', 'curbweight']]

# Add a constant
X_2c = sm.add_constant(X_2)
```

```
# Model2

lr_2 = sm.OLS(y, X_2c).fit()
lr_2.params
print(lr_2.summary())

X_3 = X[['horsepower', 'curbweight', 'enginesize']]

# Add a constant
X_3c = sm.add_constant(X_3)

# Model3

lr_3 = sm.OLS(y, X_3c).fit()
lr_3.params
print(lr_3.summary())
```

```
# Linear Regression model using backward elimination
# Finding significant variable using RFE with hyperparameter = 15

lm = LinearRegression()
lm.fit(X, y)

rfe = RFE(lm, 15)
rfe = rfe.fit(X, y)

list(zip(X.columns,rfe.support_,rfe.ranking_))

# Selecting the variables which are in support (=True,1)

col_sup = X.columns[rfe.support_]
col_sup

# X dataframe with RFE selected variables

X_rfe = X_[col_sup]

# Adding a constant variable
import statsmodels.api as sm
X_rfec = sm.add_constant(X_rfe)

#Model1
lm_rfe = sm.OLS(y,X_rfec).fit()

#Summary
print(lm_rfe.summary())
```

```
# VIF of the features

vif = pd.DataFrame()
vif['Features'] = X_rfe.columns
vif['VIF'] = [variance_inflation_factor(X_rfe.values, i) for i in range(X_rfe.shape[1])]
vif['VIF'] = round(vif['VIF'], 2)
vif = vif.sort_values(by = "VIF", ascending = False)
vif

# Dropping highly correlated variables and insignificant variables

X_rfe1 = X_rfe.drop('cylindernumber_six', 1,)

# Adding a constant variable and Build a second fitted model

X_rfe1c = sm.add_constant(X_rfe1)
lm_rfe1 = sm.OLS(y, X_rfe1c).fit()

#Summary of linear model
print(lm_rfe1.summary())

# VIF values
vif = pd.DataFrame()
vif['Features'] = X_rfe1.columns
vif['VIF'] = [variance_inflation_factor(X_rfe1.values, i) for i in range(X_rfe1.shape[1])]
vif['VIF'] = round(vif['VIF'], 2)
vif = vif.sort_values(by = "VIF", ascending = False)
vif

# Dropping highly correlated variables and insignificant variables

X_rfe2 = X_rfe1.drop('cylindernumber_twelve', 1,)

#Similarly upto model 8 we have applied same code

# Dropping highly correlated variables and insignificant variables

X_rfe9 = X_rfe8.drop(‘carbody_wagon', 1,)

# Adding a constant variable and Build a sixth fitted model
X_rfe9c = sm.add_constant(X_rfe9)
lm_rfe9 = sm.OLS(y, X_rfe9c).fit()

#Summary
print(lm_rfe9.summary())

# VIF values
```

```
vif = pd.DataFrame()
vif['Features'] = X_rfe9.columns
vif['VIF'] = [variance_inflation_factor(X_rfe9.values, i) for i in range(X_rfe9.shape[1])]
vif['VIF'] = round(vif['VIF'], 2)
vif = vif.sort_values(by = "VIF", ascending = False)
vif

# Dropping highly correlated variables and insignificant variables

X_rfe10 = X_rfe9.drop('carbody_sedan', 1,)

# Adding a constant variable and Build a sixth fitted model
X_rfe10c = sm.add_constant(X_rfe10)
lm_rfe10 = sm.OLS(y, X_rfe10c).fit()

# Create a dataframe that will contain the names of all the feature variables and their respective VIFs
vif = pd.DataFrame()
vif['Features'] = X_rfe10.columns
vif['VIF'] = [variance_inflation_factor(X_rfe10.values, i) for i in range(X_rfe10.shape[1])]
vif['VIF'] = round(vif['VIF'], 2)
vif = vif.sort_values(by = "VIF", ascending = False)
vif

#Summary of linear model
print(lm_rfe10.summary())
```

6. REFERENCES:

[1] https://www.investopedia.com/terms/v/variance-inflation-factor.asp
[2]https://towardsdatascience.com/feature-selection-in-python-recursive-feature-elimination-19f1c39b8d15
[3]https://www.statisticshowto.datasciencecentral.com/variance-inflation-factor/