Contents

1. ABSTRACT

The automobile industry is today most profitable industry. There are new brands and new features were emerging continuously in the preference of customers such as price segment, mileage, … etc. Further competition is heating up with host of new players coming in and global manufacturers. This analysis and visualization of the automobile dataset will be helpful for the existing and new entrant car manufacturing companies to find out the customer expectations and the current analysis of various thousands of variants of vehicles that are running in the market currently. This project presents various levels of visualizations using various histograms, bar plots and box plots. And data analysis of consumer automobiles to get a proper understanding of consumer buying and pricing behavior of vehicles that are currently in market to predict prices of future cars based on their other attributes.

2. INTRODUCTION

The automobile data analysis includes a dataset introduced from the University of California Irvine Machine Learning Repository UCI and refined from Kaggle. According to UCI (1985), the attributes consist of three different types of entities: (a) the model and specification of an auto, which includes the characteristics, (b) the personal insurance, (c) its normalized losses in use as compared to other cars. The data set source for this model collected from Insurance collision reports, personal insurance, and car models. According to Kaggle, there are 24 data attributes in this model descript the data set model from different angles. The objective of this report is to perform exploratory data analysis to find the primary relationships between features, which include univariate analysis, Bivariate analysis and multivariate analysis which includes finding the maximum and minimum, such as the weight, length, horsepower, and price. Moreover, we are performing a prediction model. The insights that could be estimated from this dataset would be feature such as price of a specific car model that could be estimated using the other attributes of that particular car model using machine learning algorithms like Linear Regression. The objective also includes the study of various attributes of the considered Indian automobile dataset and finding the relationship or statistically, finding the correlation between them and visualizing the findings.

The reason for choosing this particular project was because of its practical applications involved in it. Many people often face the problem of pricing vehicles while they are selling it online. Thus, a prediction model capable of pricing of a particular model of a car can be useful when an owner wants to sell their vehicle. Also, these factors help with that.

2.1 Attribute information: -

1. symbolling: -3, -2, -1, 0, 1, 2, 3.
2. make: alfa-romero, audi, bmw, chevrolet, dodge, honda, isuzu, jaguar, mazda, mercedes-benz, mercury, mitsubishi, nissan, peugot, plymouth, porsche, renault, saab, subaru, toyota, volkswagen, volvo

3. fuel-type: diesel, gas.
4. aspiration: std, turbo.
5. num-of-doors: four, two.
6. body-style: hardtop, wagon, sedan, hatchback, convertible.
7. drive-wheels: 4wd, fwd, rwd.
8. engine-location: front, rear.
9. wheel-base: continuous from 86.6 120.9.
10. length: continuous from 141.1 to 208.1.
11. width: continuous from 60.3 to 72.3.
12. height: continuous from 47.8 to 59.8.
13. curb-weight: continuous from 1488 to 4066.
14. num-of-cylinders: eight, five, four, six, three, twelve, two.
15. engine-size: continuous from 61 to 326.
16. fuel-system: 1bbl, 2bbl, 4bbl, idi, mfi, mpfi, spdi, spfi.
17. bore: continuous from 2.54 to 3.94.
18. stroke: continuous from 2.07 to 4.17.
19. compression-ratio: continuous from 7 to 23.
20. horsepower: continuous from 48 to 288.
21. peak-rpm: continuous from 4150 to 6600.
22. city-mpg: continuous from 13 to 49.
23. highway-mpg: continuous from 16 to 54.
24. price: continuous from 5118 to 45400.

3. METHODOLODGY

The project is about data analysis on Automobile industry, The entire project divided into two parts which are visualization and the data analysis parts of the project. The visualization part of the project deals with the various plotting of attributes while the data analysis part of the project deals with finding the relationship between various attributes in the dataset.

First the data selected from machine learning repository. Which contains missing values, then imported the same refined data from Kaggle which does not contain missing values for more accurate results. Imported data into python and split the make name and transformed into company name of vehicles for analyzing. After that checked for duplicates.

The visualization part consists of univariate analysis, analyzing the data in perspective of a single attribute then with bivariate analysis, analysis using two attributes and then with multivariate which deals with more than two attributes at the same time. Here the attribute's distributions are visualized using bar plots, histogram, box plots, scatter plots etc. The bivariate analysis is done using scatter plots, box plots. Multivariate analysis done by correlation heat map.

The data analysis is performed on the automobile dataset utilizing machine learning algorithms in order to study the various relationships between attributes of the considered automobile dataset and attempts to consolidate the findings of the relationship between the attributes or statistically, finding the correlation between them and visualizing the findings. Of these features some of them might be a redundant and might be a good contributor to the prediction model and the task of eliminating such attributes also shall be considered. The result of finding this relationship between various attributes of a vehicle will provide useful insights in building in a prediction model capable of predicting the

price of a vehicle based on the other parameters. Here Performed two types of models. First one is Multiple regression model. For Multiple regression model here checked various influencing factor (VIF). And getting into a decent model. The Second model done here is KNN algorithm.

All this visualization and analysis is done through Python programming. For python here imported so many existing functions for that. Given below is the existing functions had used: -

- Numpy
- Pandas
- Matplotlib.pyplot
- Seaborn
- Sklearn.model_selection
- Sklearn.preprocessing
- sklearn.feature_selection
- sklearn.linear_model
- statsmodels.api
- statsmodels.stats.outliers_influence
- sklearn.metrics
- sklearn.neighbors

4. Data Analysis

The data about analysis of automobiles. In previous section 2 and section 3 have described the characteristics of data and methodologies using in analysis of car data. The section 4 is about Exploratory data analysis and Model building.

4.1 Exploratory Data Analysis

Exploratory data analysis is an integral part data analysis project. The EDA (Exploratory Data Analysis) contains Univariate analysis, Bivariate analysis and multivariate analysis of car data.

Table 1

Index	symboling	wheelbase	carlength	carwidth	carheight	curbweight	enginesize	boreratio	stroke	compressionratio	horsepower	peakrpm	citympg	highwaympg	price
count	205	205	205	205	205	205	205	205	205	205	205	205	205	205	205
mean	0.83415	98.757	174.05	65.908	53.725	2555.6	126.91	3.3298	3.2554	10.143	104.12	5125.1	25.22	30.751	13277
std	1.2453	6.0218	12.337	2.1452	2.4435	520.68	41.643	0.27084	0.3136	3.972	39.544	476.99	6.5421	6.8864	7988.9
min	-2	86.6	141.1	60.3	47.8	1488	61	2.54	2.07	7	48	4150	13	16	5118
25%	0	94.5	166.3	64.1	52	2145	97	3.15	3.11	8.6	70	4800	19	25	7788
50%	1	97	173.2	65.5	54.1	2414	120	3.31	3.29	9	95	5200	24	30	10295
75%	2	102.4	183.1	66.9	55.5	2935	141	3.58	3.41	9.4	116	5500	30	34	16503
max	3	120.9	208.1	72.3	59.8	4066	326	3.94	4.17	23	288	6600	49	54	45400

The table 1 shows overall summary such as mean, standard deviation etc… of continuous variables. Analysis is done, price as target variable. Here from table 1, shows average price of cars is 13277.

Table 2

```
count      205.000000
mean     13276.710571
std       7988.852332
min       5118.000000
25%       7788.000000
50%      10295.000000
75%      16503.000000
85%      18500.000000
90%      22563.000000
100%     45400.000000
max      45400.000000
Name: price, dtype: float64
```

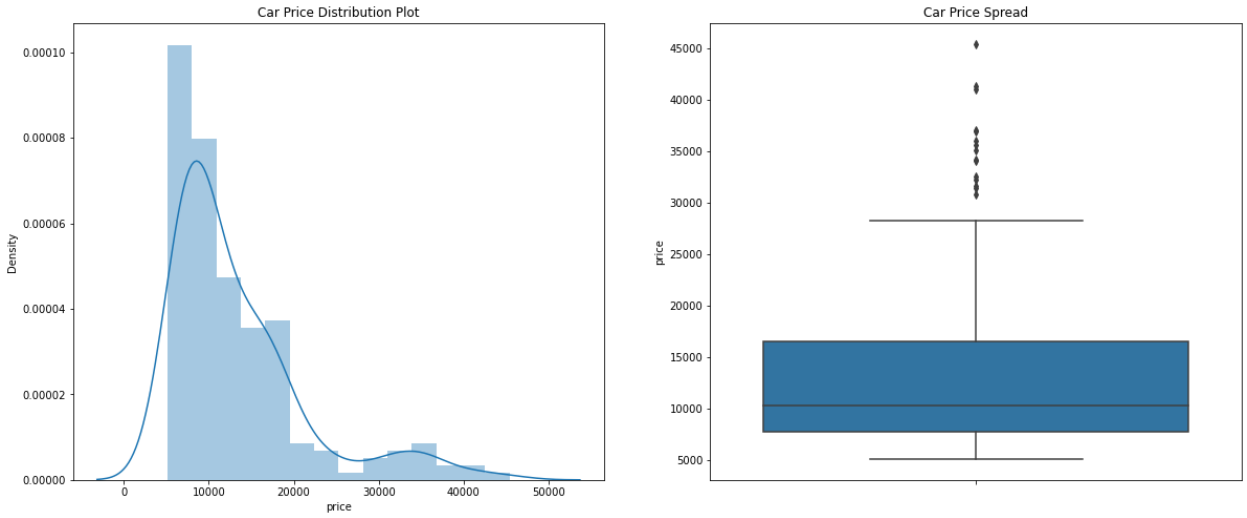

Figure 1

From figure 1 the plot seemed to be right-skewed, meaning that the most prices in the dataset are low (Below 15,000). Plot shows significant difference between the mean and the median of the price distribution. The data points are far spread out from the mean, which indicates a high variance in the car prices. (85% of the prices are below 18,500, whereas the remaining 15% are between 18,500 and 45,400.)

4.1.2. Univariate Analysis and Bivariate Analysis

The project is about data analysis on automobiles. Here we are taken univariate analysis of company's brand, fuel type, and car type which is most significant by customers while buying cars.

From the figure 2, multiple bar diagram shows that Toyota is the most favored car company. Nissan, Mazda, Mitsubishi are in second, third and fourth position so on. Jaguar, Chevrolet, Alfa, Renault, and Mercury are least favored car companies.

Figure 3 shows, Fuel type is one of the most significant attributes while purchasing a car. Day by day fuel price is increasing. From figure 3 shows that number of gas fuels are more than diesel. That is, Gas more favored for customers. Because of Gas vehicles are cheaper.

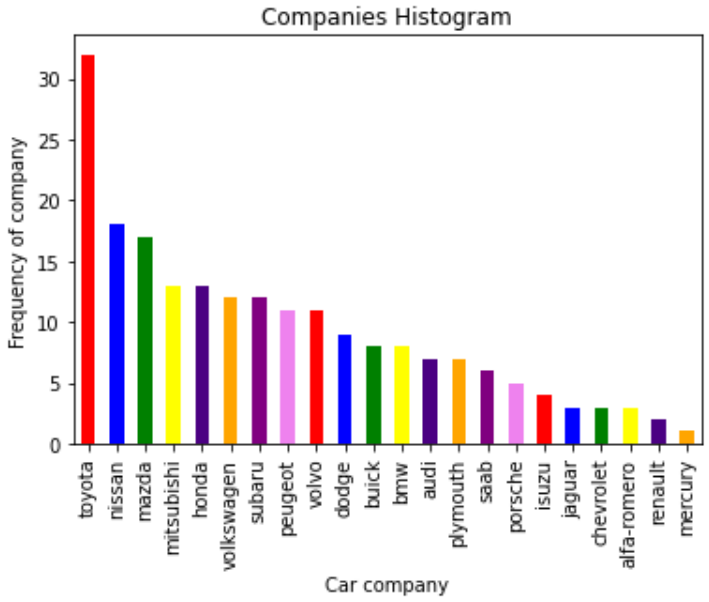

Figure 2

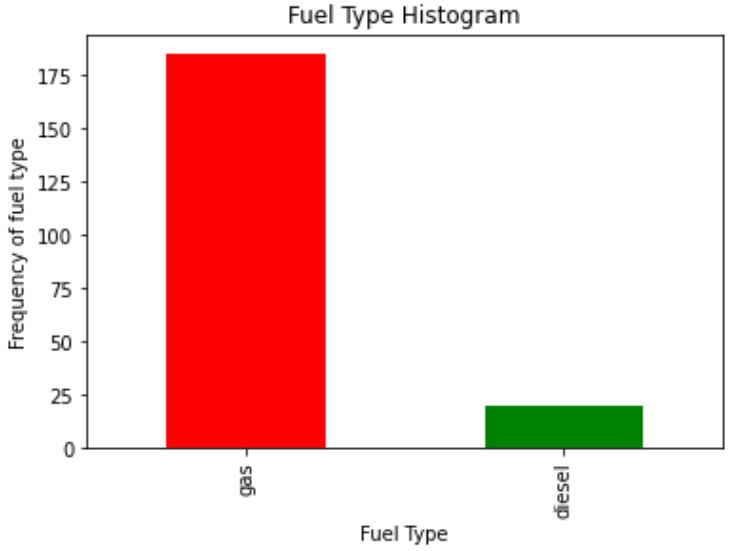

Figure 3

Figure 3 shows, Fuel type is one of the most significant attributes while purchasing a car. Day by day fuel price is increasing. From figure 3 shows that number of gas fuels are more than diesel. That is, Gas more favored for customers. Because of Gas vehicles are cheaper.

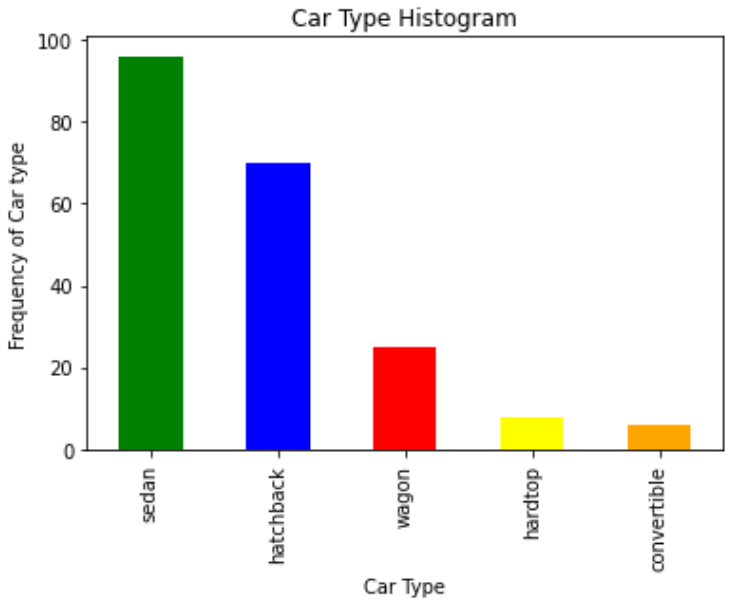

Figure 4

From figure 4, While considering car type, Sedan is the top car preferred. Because they are smaller and low to the ground, sedans are less likely to tip and they usually have a smaller turning radius than some of their larger counterparts like full-size SUVs and trucks. In fact, many sedans are turned into performance vehicles because of their excellent handling.

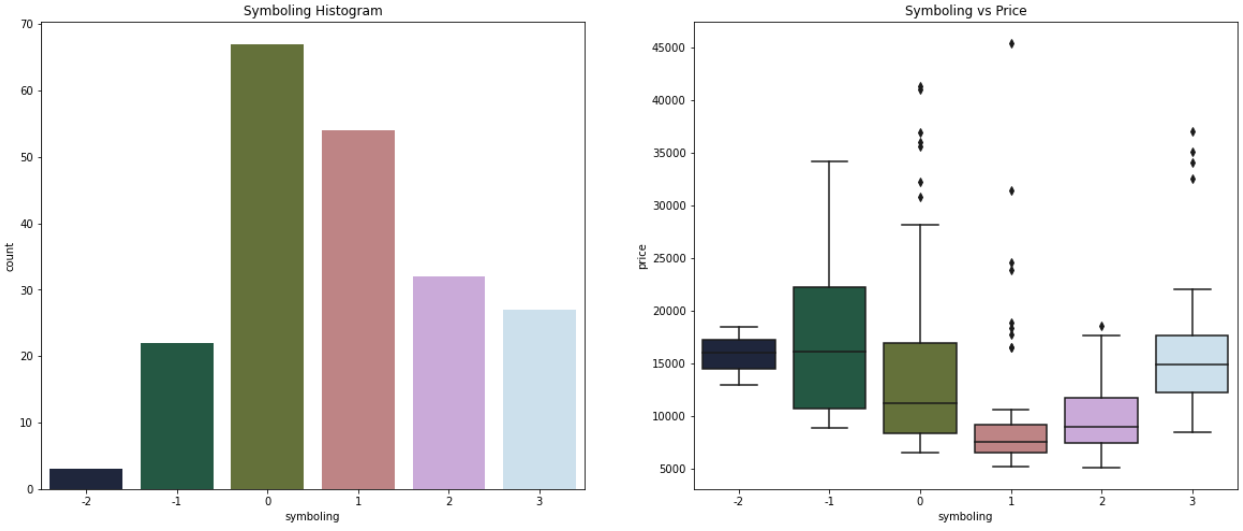

Figure 5

Figure 5 shows symbolling histogram and box plot between symbolling and price. Symbolling with 1 and 0 is the most sold vehicles. It means that it is average safety-maintained vehicle. It seems that the symbolling with 0 and 1 values have high number of rows (i.e. They are most sold). The cars with -1 symbolling seems to be high priced (as it makes sense too, insurance risk rating -1 is quite good). But it seems that symbolling with 3 value has the price range similar to -2 value. There is a dip in price at symbolling.

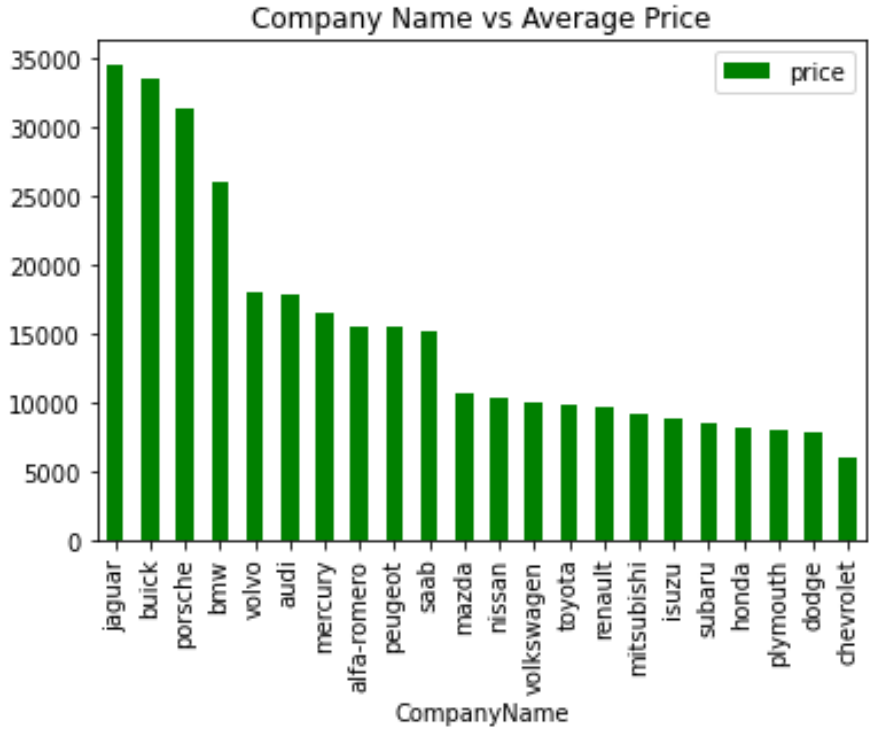

Figure 6

Figure 6 shows that the bivariate analysis of company name to price, we can see that jaguar is the most priced vehicle. That is one of the reason Jaguar is least preferred vehicle by customers. From figure 3 got Toyota is the most preferred company, from figure 6 can say one of the reasons for that it is not much costly. It is one of the most affordable brands.

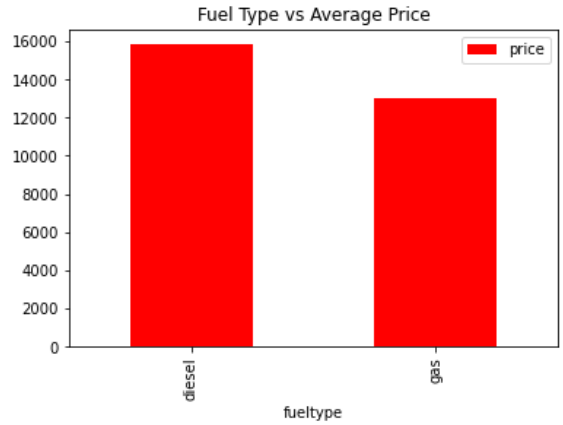

Figure 7

From figure 3 it is analyzed that gas fuel type is more preferred by customers than diesel. Here figure 7 shows the reason for that. Gas vehicle is cheaper while comparing with diesel vehicle.

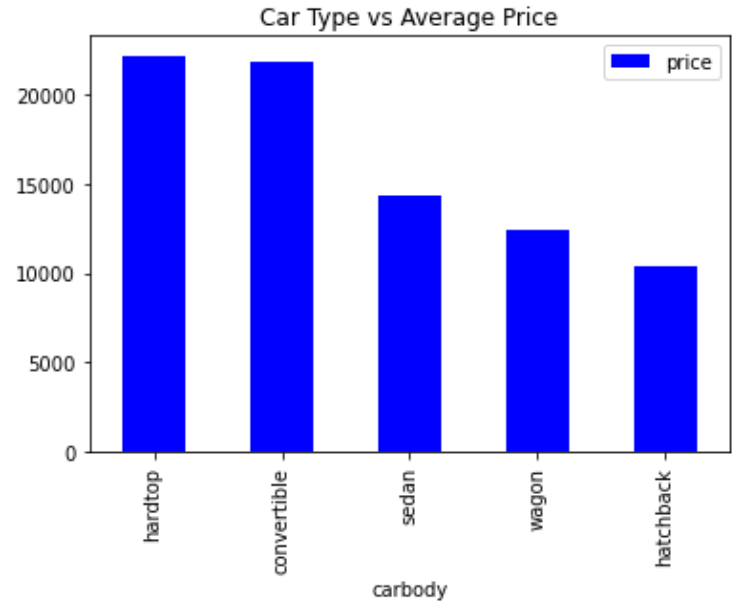

Figure 8

In figure 4 it is analyzed that sedan is most preferred and hardtop and convertible less preferred car type by customers. Figure 8 shows main reason for that, hardtop and convertible car type is highly priced. Its average price is greater than 20000. But in sedan average

price is up to 15000. In figure 1 shows, car price univariate analysis which is most preferred car price.

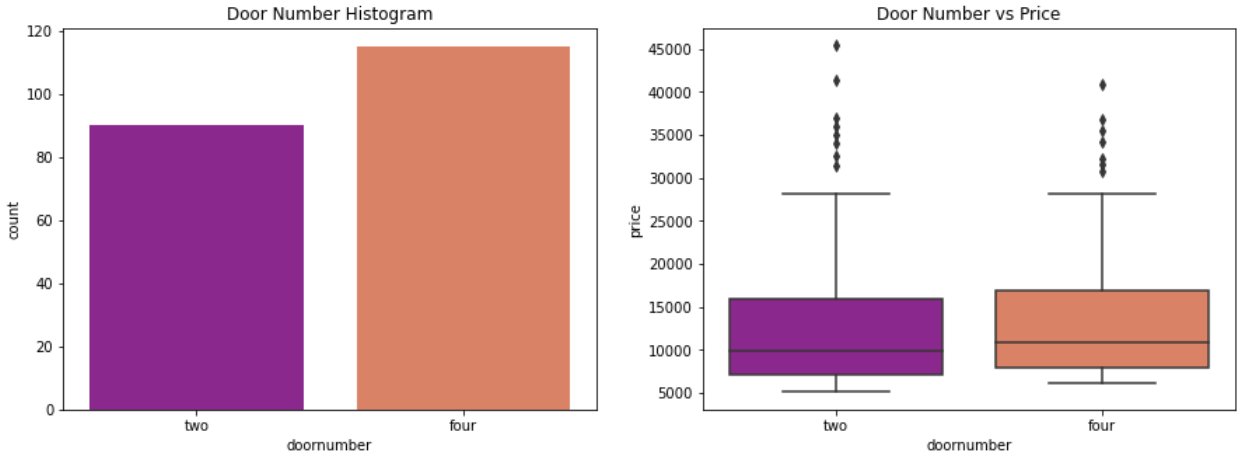

Figure 9

Figure 9 interpreting that door number four is more than two. But when it comparing to price it does not affect. Which means that the door number is selected by size of members in customers families…etc.

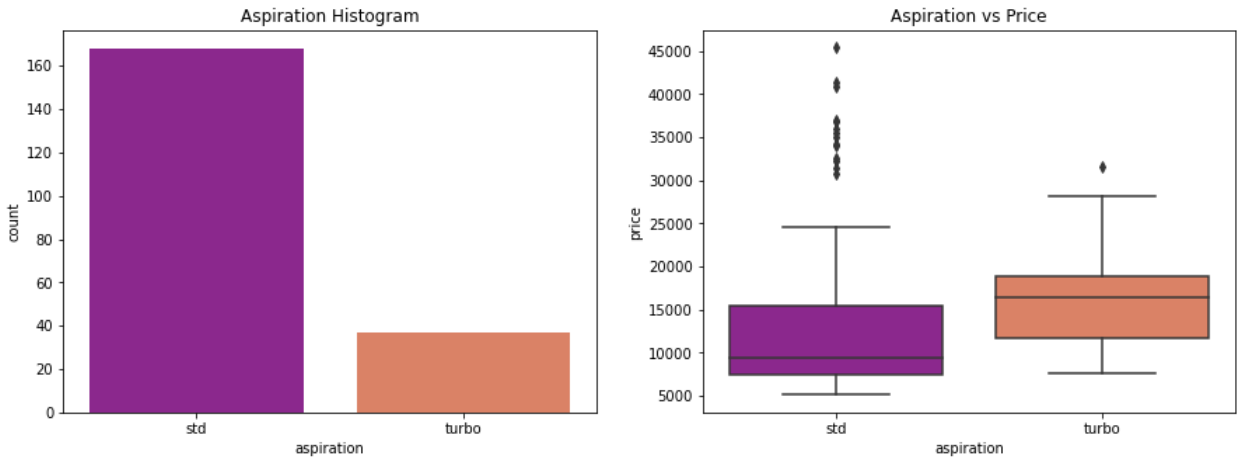

Figure 10

From figure 10 shows that std engine is highly sold than turbo because std engine is cheaper than turbo engine. Turbocharged engines differ from standard engines in that they make use of wasted exhaust gases to pull more air into the intake valve. While naturally-

aspirated engines rely on natural air pressure to draw air into the engine, turbos speed up this process, producing power more economically. People are not severely looking to engine features. Because turbo engines are producing more economically.

Figure 11 shows the univariate of Engine location, cylinder number, fuel type, drive wheel type and bivariate analysis of this variable comparing with price. From figure 11 the plot of engine location it is analyzed that engine location front is most sold. Because very cheaper than engine location rear. The average price of engine location is front and rear is up to 18000 and more than 30000. Which is almost doubling.

From figure 11 it is analyzed that In cylinder number, most car sold four cylinders vehicle which is less costly while comparing others are highly priced except three. But there is no much difference between three and four. In four that can store more fuels than three those are almost equal price.

From figure 11 it is analyzed that fuel type mpfi and 2bbl are most common type of fuel systems. mpfi and idi having the highest price range. But there are few data for other categories to derive any meaningful inference.

From Figure 11 it is analyzed that drive wheel plot front wheel driving car is most sold and also it is cheaper than others and it is more comfortable too.

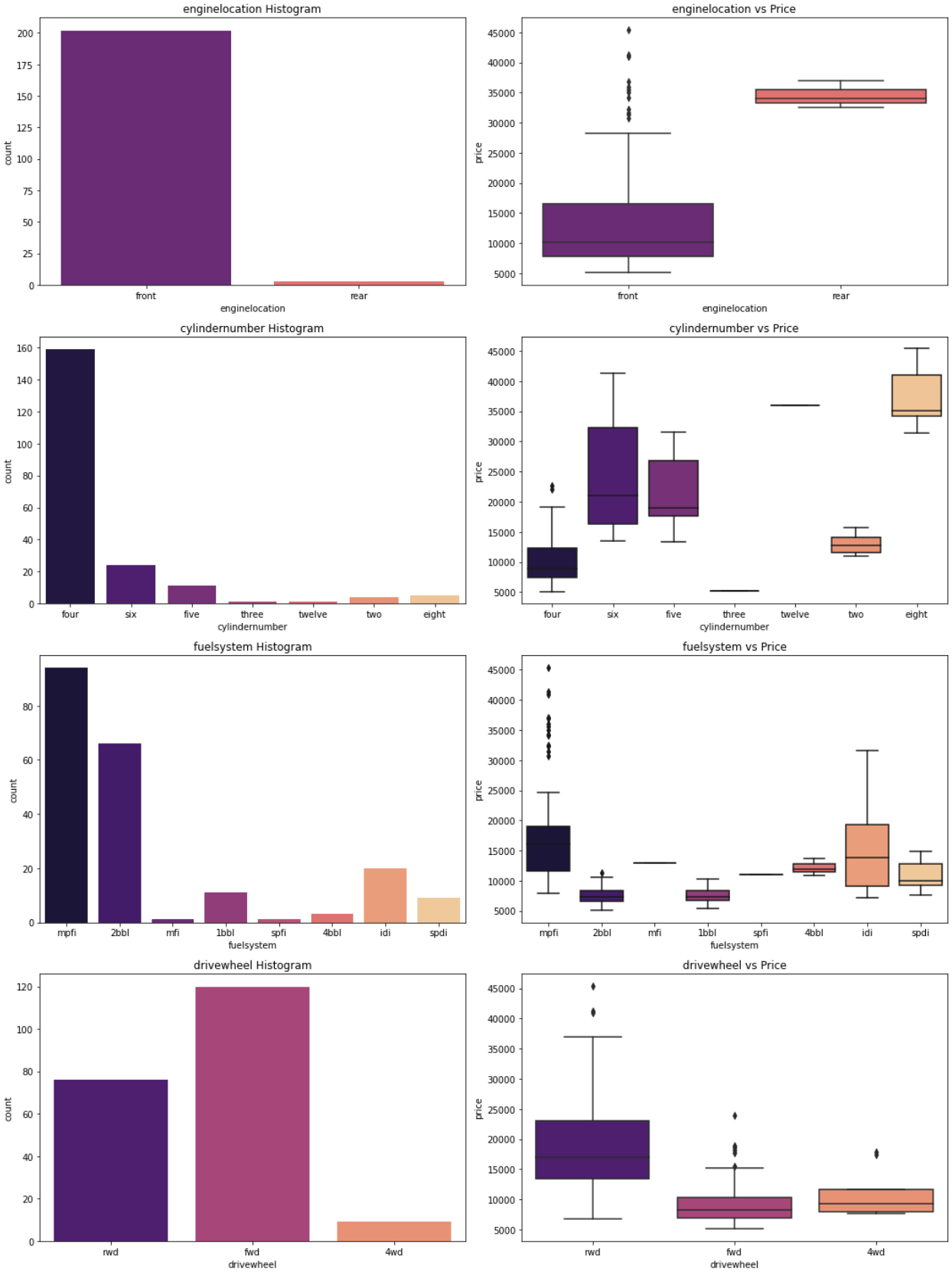
enginelocation Histogram
count
enginelocation
front
rear
enginelocation vs Price
price
enginelocation
front
rear
cylindernumber Histogram
count
cylindernumber
four
six
five
three
twelve
two
eight
cylindernumber vs Price
price
cylindernumber
four
six
five
three
twelve
two
eight
fuelsystem Histogram
count
fuelsystem
mpfi
2bbl
mfi
1bbl
spfi
4bbl
idi
spdi
fuelsystem vs Price
price
fuelsystem
mpfi
2bbl
mfi
1bbl
spfi
4bbl
idi
spdi
drivewheel Histogram
count
drivewheel
rwd
fwd
4wd
drivewheel vs Price
price
drivewheel
rwd
fwd
4wd

Figure 11

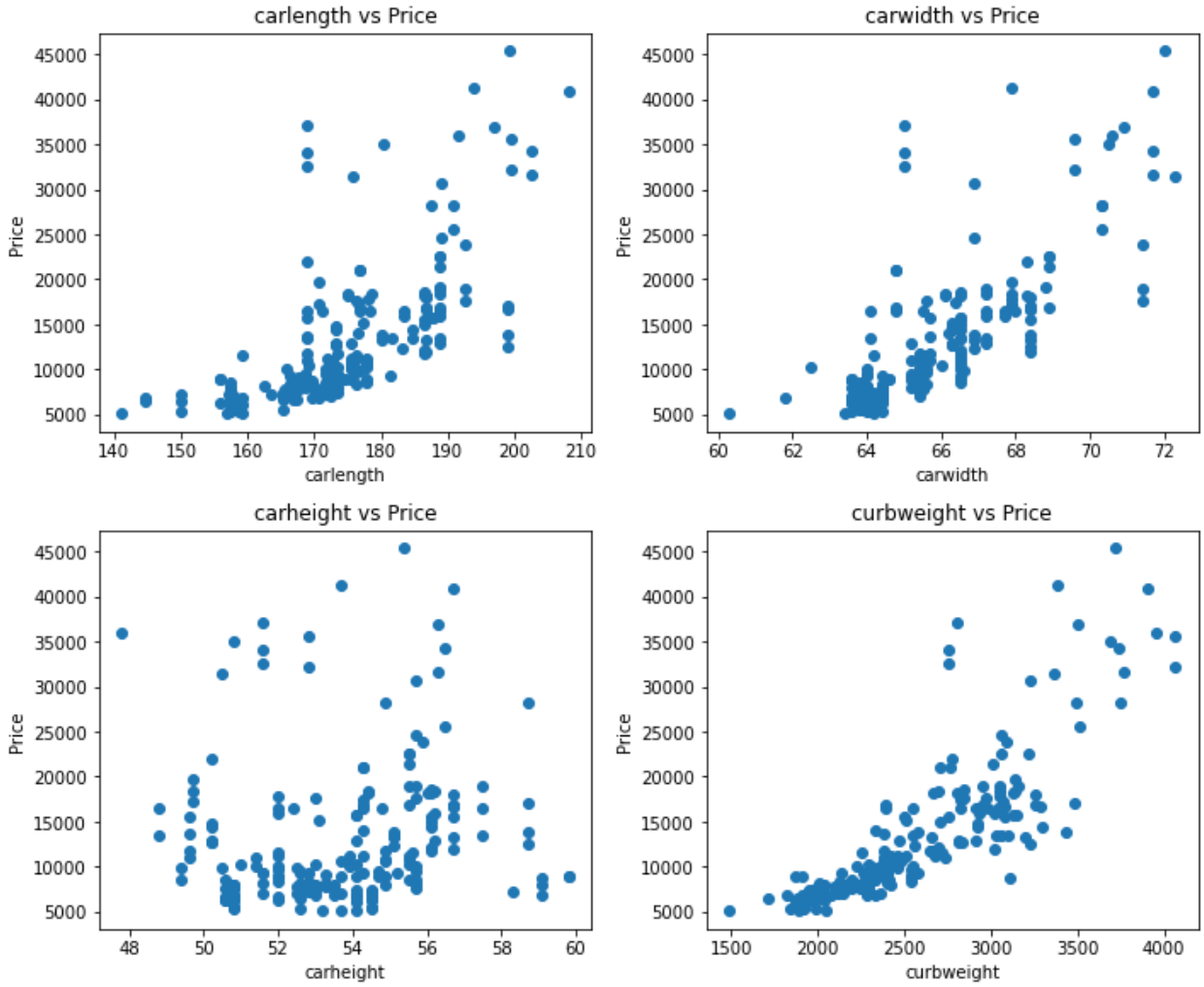

Figure 12

Figure 12 interpreted that Car length, Car width and Curb weight shows positive correlation with price. Which mean when these features increasing price also increasing. Car height does not shows any particular trend in above scatter plot.

Figure 13,14,15 interpreting that Engine size, bore ratio, horsepower, wheelbase - seem to have a significant positive correlation with price. City mpg, highway mpg - seem to have a significant negative correlation with price in figure 15 scatter plots.

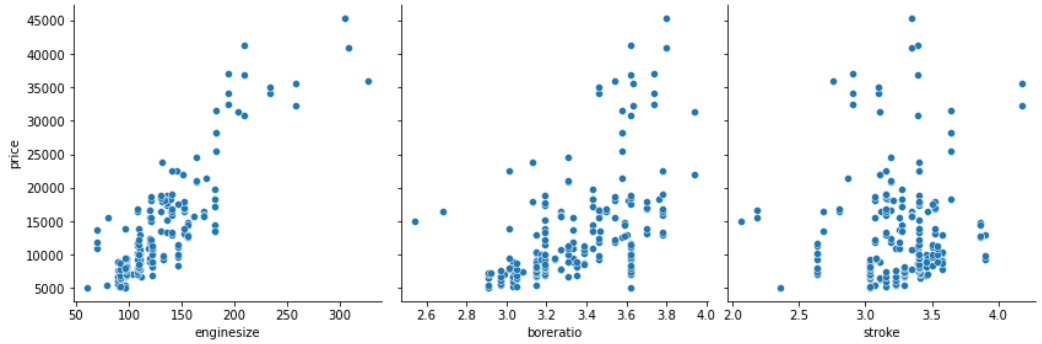

Figure 13

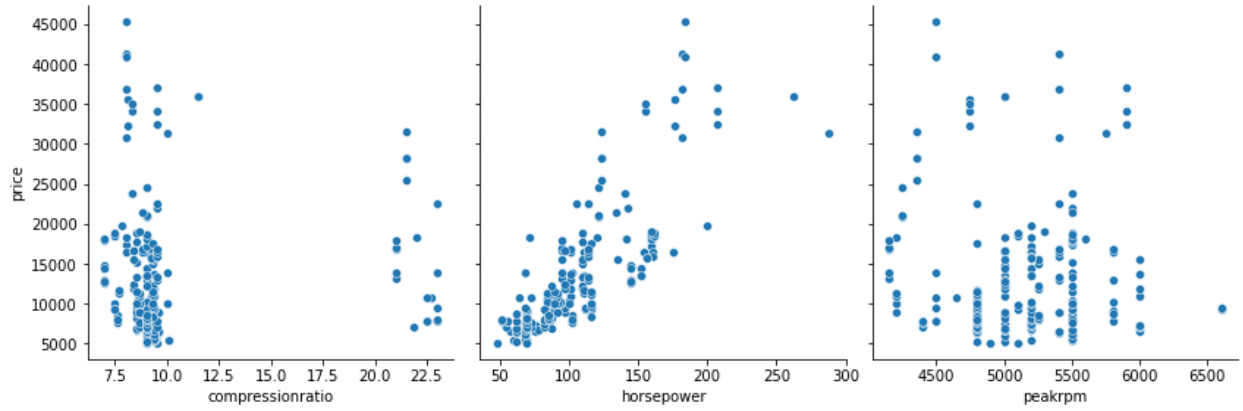

Figure 14

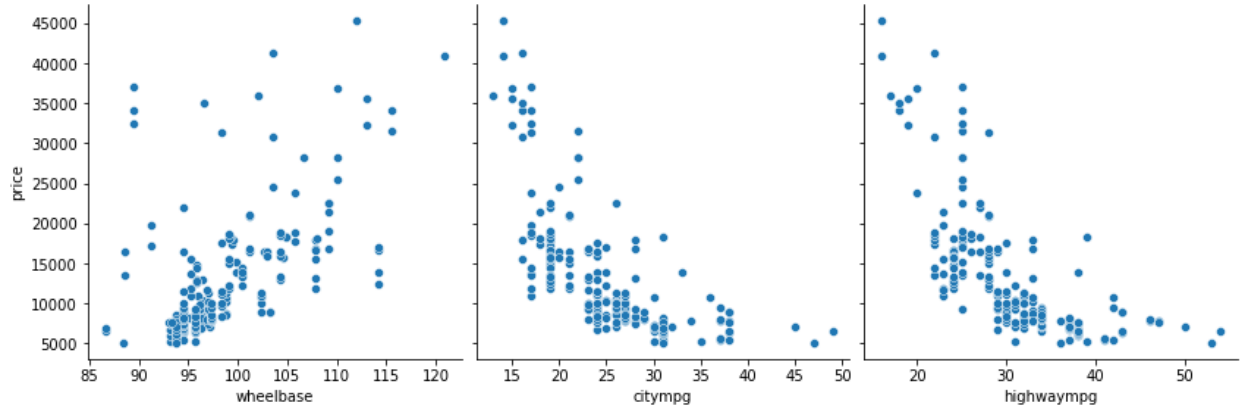

Figure 15

The significant variables after visual analysis are given below: -

1)Fuel type

2)Car Body

3)Aspiration

4)Cylinder Number

5)Drive wheel

6)Curb weight

7)Car Length

8)Car width

9) Engine Size

10) Bore ratio

11) Horse Power

12) Wheel base

From figure 16 shows that correlation plots of every continuous variable. It is already analyzed that car length, width and curb weight is highly correlated. And city-mpg and highway mpg is negatively correlated.

Figure 17 shows the heatmap of correlation for easy understanding of figure 16. The thickness of colors depends correlation. High thickness means high correlation and low thickness means low correlation.

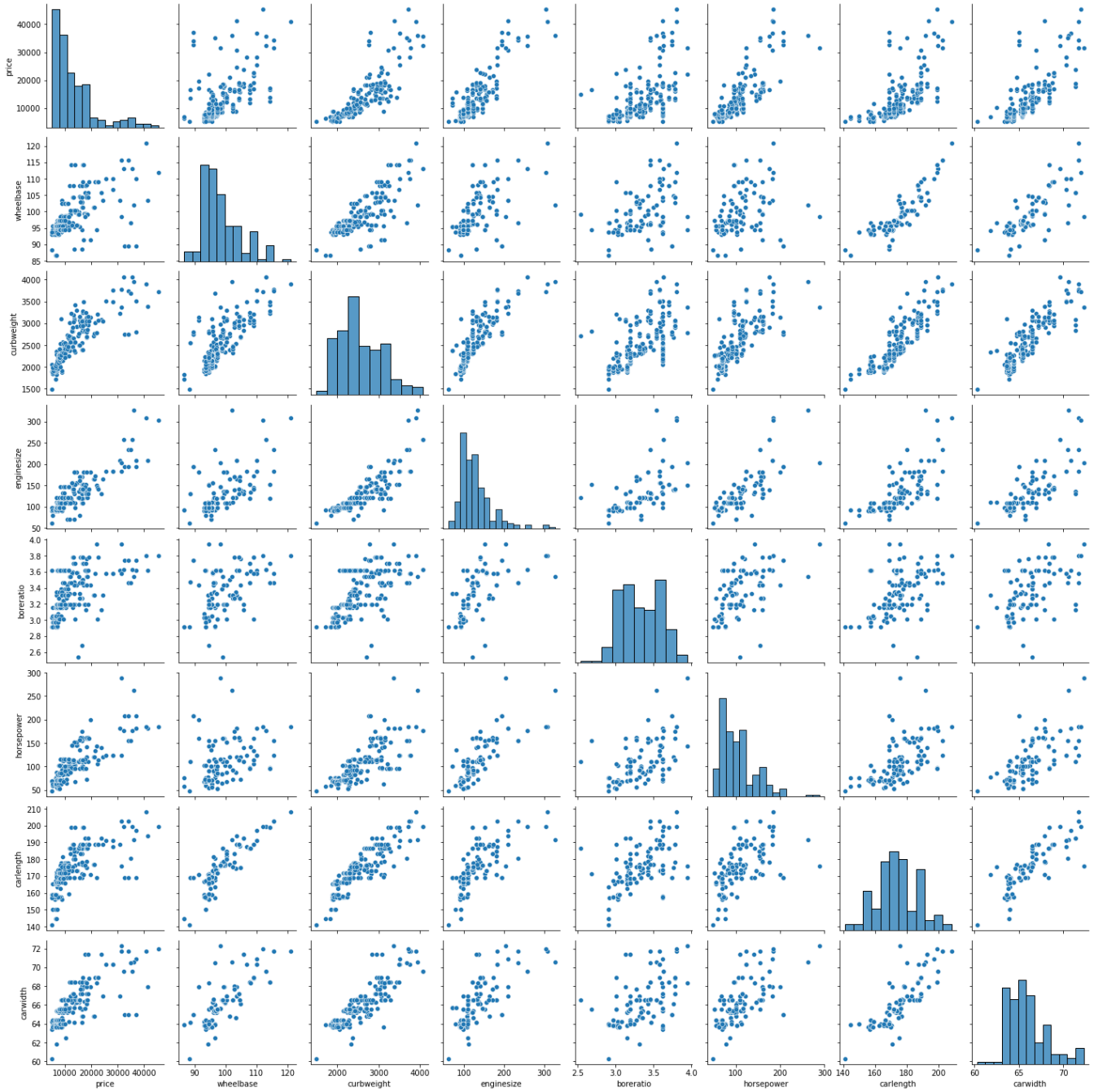
price
wheelbase
curbweight
enginesize
boreratio
horsepower
carlength
carwidth

Figure 16

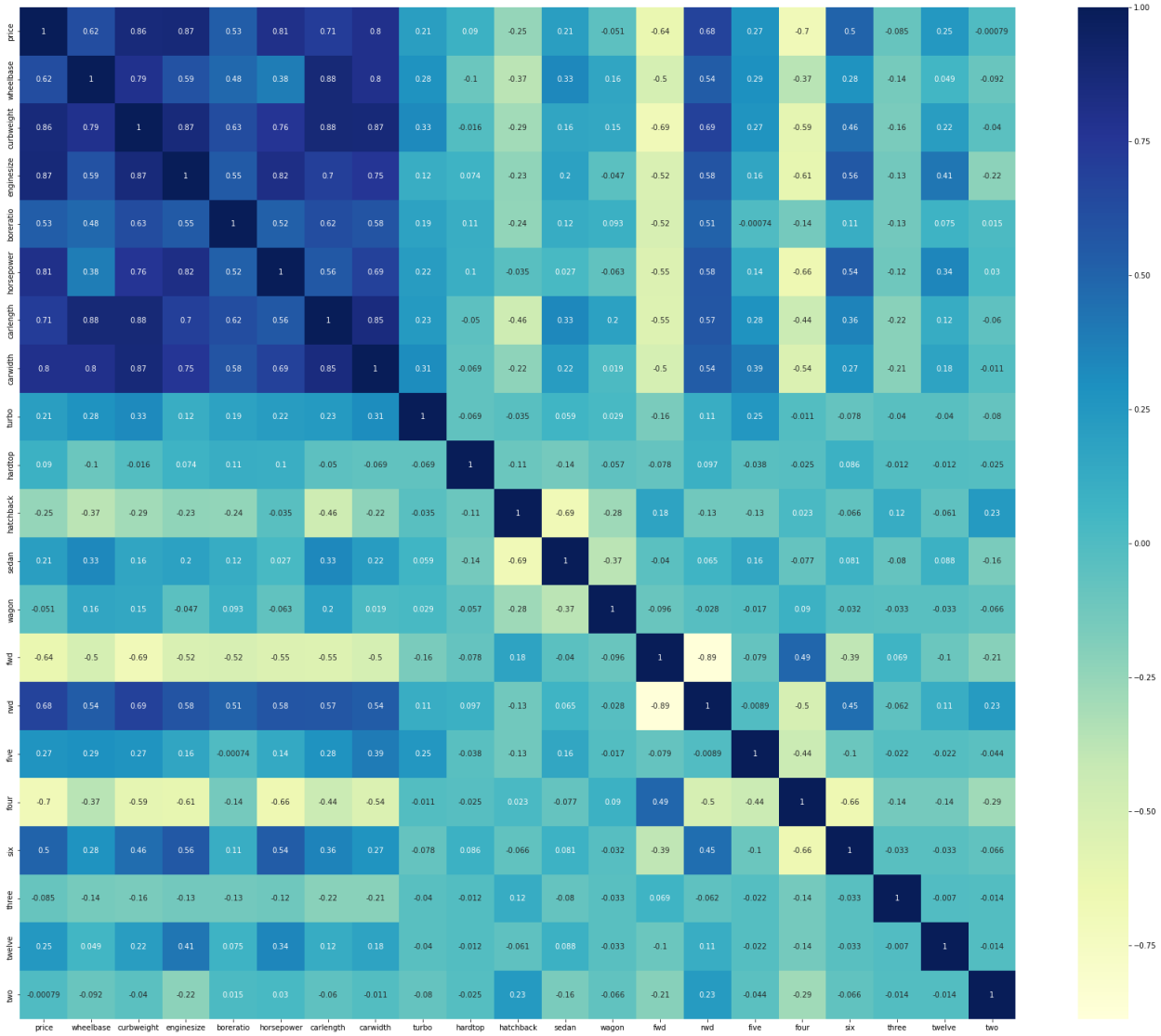
price wheelbase curbweight enginesize boreratio horsepower carlength carwidth turbo hardtop hatchback sedan wagon fwd rwd five four six three twelve two
1.00 0.75 0.50 0.25 0.00 −0.25 −0.50 −0.75

Figure 17

4.2 Model Building

4.2.1 Multiple Regression Model: -

In section 4.1 and 4.1.2 interpreted the automobile data visualization such as univariate analysis, bivariate analysis, multivariate analysis and correlation heatmap. It is interpreted that 12 variables out of 23 variables are sufficient for model buildings.

In this section 4.2 are building the regression model with price as targeted variable. Now created dummy variables for categorical dependent variable. Then build multiple regression model. At first the models show some of the variables such as **wheel base, car length, and wagon** are not significantly affecting the price according to p-value. Figure 18 shows the built model by dropping these variable one by one. At last, by dropping these variables most decent model received with R squared value is 0.854 and adjacent R squared value is 0.846. In table 3 shows checking the variance influence factor too. Which is also decent.

Table 3

```
In [116]: checkVIF(X_train_new2)
Out[116]:
      Features   VIF
0        const  9.88
1   enginesize  5.70
2   horsepower  3.76
3     carwidth  2.77
7          two  1.31
6       twelve  1.29
4    hatchback  1.19
5        three  1.06
```

OLS Regression Results

Dep. Variable:	price	R-squared:	0.854
Model:	OLS	Adj. R-squared:	0.846
Method:	Least Squares	F-statistic:	112.4
Date:	Wed, 29 Mar 2023	Prob (F-statistic):	3.98e-53
Time:	21:49:48	Log-Likelihood:	154.29
No. Observations:	143	AIC:	-292.6
Df Residuals:	135	BIC:	-268.9
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.1508	0.022	-6.775	0.000	-0.195	-0.107
enginesize	0.8340	0.110	7.605	0.000	0.617	1.051
horsepower	0.2644	0.083	3.178	0.002	0.100	0.429
carwidth	0.2611	0.064	4.071	0.000	0.134	0.388
hatchback	-0.0486	0.016	-2.983	0.003	-0.081	-0.016
three	0.2003	0.087	2.295	0.023	0.028	0.373
twelve	-0.2898	0.097	-2.998	0.003	-0.481	-0.099
two	0.1963	0.049	3.993	0.000	0.099	0.294

Omnibus:	20.714	Durbin-Watson:	1.849
Prob(Omnibus):	0.000	Jarque-Bera (JB):	55.375
Skew:	0.511	Prob(JB):	9.45e-13
Kurtosis:	5.872	Cond. No.	23.2

Figure 18

4.2.2 Residual Analysis and Accuracy

The Model had built for car data and received good R- squared value in section 4.2. Here section 4.2.2 is analyzing in error term in the model whether it follows normal distribution or not.

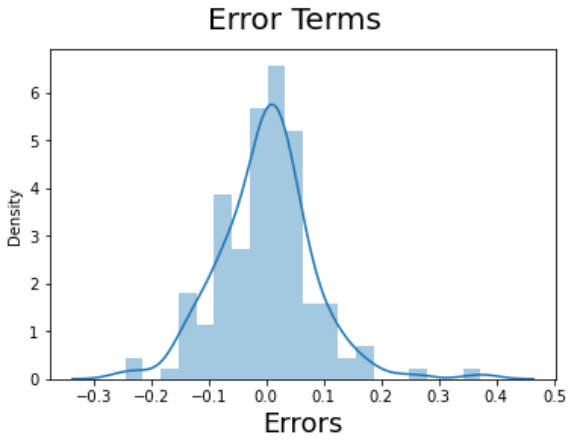

Figure 19

Figure 19 shows residual analysis Error term is approximately normally distributed which means it is fulfilled linear modelling assumptions.

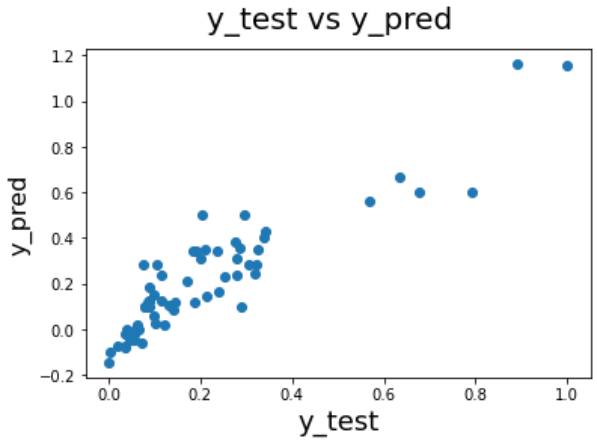

Figure 20

Figure 20 shows the scatter plot between predicted price values and test price value. There is an accuracy of 74%. *R-squared and Adjusted R-squared (extent of fit)* - 0.854 and 0.846. *p-values* - p-values for all the coefficients seem to be less than the significance level of 0.05. - meaning that all the predictors are statistically significant.

Interpretation: -

From the figure 18 The model interpreting that price depends on engine size, horse power, car width, hatchback, cylinder number of three, two, and twelve.

4.3 K Nearest Neighborhood: -

Section 4.2 build multiple regression model and checked the accuracy of the model. In section 4.3 concentrate on building KNN.

4.3.1 KNN: -

In figure 21 shows the predicted prices of KNN;

```
array([23842. , 18641. ,  8329.8, 10365.6, 21066. ,  7444.4,  8429. ,
        7264.6,  9488. ,  8656.4, 18641. ,  9945.4, 15617. , 10176.4,
       34784. ,  5956.4,  6441.4, 13741.2,  8329.8,  8216.6, 10206. ,
       19165.8,  9208.2,  6296.8,  7417.5, 28987. , 20774.6, 14522.6,
        7880.8, 15388. , 20705. ,  5987.8,  7670.6, 23842. ,  7754.2,
       24104.6, 12822. , 12344.2,  7408.9, 13741.2,  9488. ])
```

Figure 21

Figure 21 shows KNN model which contain variables such as wheel base, curb weight, engine size, bore ratio, horsepower, car length, car width, fuel type, aspiration ratio, car body, drive wheel, cylinder number.

```
wheelbase : 0.03486666621084714
curbweight : 0.0568290601211652
enginesize : 0.10594647563071202
boreratio : 0.01760345934895422
horsepower : 0.04482437650009914
carlength : 0.04828257685624919
carwidth : 0.07598817826640523
fueltype_gas : -0.008746330832836092
aspiration_turbo : 0.0166526760866025
carbody_hardtop : -0.027498916723726775
carbody_hatchback : 0.047631914523212235
carbody_sedan : 0.055891870259644406
carbody_wagon : 0.012784351336653565
drivewheel_fwd : 0.019465162854723984
drivewheel_rwd : 0.027029323137758778
cylindernumber_five : 0.04785272625118323
cylindernumber_four : 0.029113164078028487
cylindernumber_six : 0.014840765801051158
cylindernumber_three : 0.0
cylindernumber_twelve : 0.0
cylindernumber_two : 0.018741947969186025
```

Figure 22

Figure 23 gives the R-squared value is 0.824096. Which is decent KNN Model.

```
In [1171]: print("R-squared:", r_squared)
R-squared: 0.8240963233632923
```

Figure 23

Interpretation: -

From figure 22 interpreted KNN shows that Engine size, horse power, car width, car length, car type sedan is highly influencing price of the car.

5. Conclusion

Thus, here visualized and derived various insights from the considered automobile dataset by performing data analysis that utilizing machine learning algorithms in Python programming language. In EDA it is performed univariate analysis, analyzing the data in perspective of a single attribute then with bivariate analysis, analysis using two attributes and then with multivariate which deals with more than two attributes at the same time presenting various levels of visualizations using bar plots, histograms, scatter plots, boxplots. The result of finding this relationship between various attributes of a vehicle will provide useful insights in building in a prediction model capable of predicting the price of a vehicle based on the other attributes. In model building it is derived regression model and studied the results, outcomes, and interpretations in addition to the methodologies to evaluate these models. Here also performed KNN method. From the data analysis it is summarized that the attributes Engine size, horsepower, car type, car cylinders are highly influencing the pricing of the cars.

6. Reference

1) https://archive.ics.uci.edu/ml/datasets/automobile
2) https://www.kaggle.com/search?q=automobile+data+in%3Adatasets
3) https://www.kia.com/dm/discover-kia/ask/what-is-the-difference-between-a-hatchback-and-a-sedan.html
4) https://motoroctane.com/news/209398-turbo-non-turbo-engine
5) https://blog.notesmatic.com/factors-affecting-vehicle-demand-and-sales-in-the-automobile-industry/

7. APPENDIX

```
import warnings
warnings.filterwarnings('ignore')
# Import the libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
#importing the data
cd C:\Users\DELL\Desktop\FInal Year Project
cars=pd.read_csv("CarPrice_Assignment.csv",)
cars.head()
cars.shape
data=cars.describe()
cars.info()
#Splitting company name from CarName column
CompanyName = cars['CarName'].apply(lambda x : x.split(' ')[0])
cars.insert(3,"CompanyName",CompanyName)
cars.drop(['CarName'],axis=1,inplace=True)
cars.head()
jp=cars
cars.CompanyName.unique()
cars.CompanyName = cars.CompanyName.str.lower()
def replace_name(a,b):

cars.CompanyName.replace(a,b,inplace=True)
replace_name('maxda','mazda')
replace_name('porcshce','porsche')
replace_name('toyouta','toyota')
replace_name('vokswagen','volkswagen')
replace_name('vw','volkswagen')
cars.CompanyName.unique()
#Checking for duplicates
cars.loc[cars.duplicated()]
cars.columns
#plots
plt.figure(figsize=(20,8))
plt.subplot(1,2,1)
plt.title('Car Price Distribution Plot')
sns.distplot(cars.price)
plt.subplot(1,2,2)
plt.title('Car Price Spread')
sns.boxplot(y=cars.price)
plt.show()
data_1=print(cars.price.describe(percentiles = [0.25,0.50,0.75,0.85,0.90,1]))
plt.figure(figsize=(10,6))
colour=["red","blue","green","yellow","indigo","orange","purple","violet"]
plt1 = cars.CompanyName.value_counts().plot(kind='bar',color=colour)
plt.title('Companies Histogram')
plt1.set(xlabel = 'Car company', ylabel='Frequency of company')
colour=['red','green']
plt1 = cars.fueltype.value_counts().plot(kind='bar',color=colour)
plt.title('Fuel Type Histogram')
plt1.set(xlabel = 'Fuel Type', ylabel='Frequency of fuel type')
colour=['green','blue','red','yellow','orange']
```

```
plt1 =
cars.carbody.value_counts().plot(kind='bar
',color=colour)
plt.title('Car Type Histogram')
plt1.set(xlabel = 'Car Type',
ylabel='Frequency of Car type')
plt.show()
#
plt.figure(figsize=(20,8))
plt.subplot(1,2,1)
plt.title('Symboling Histogram')
sns.countplot(cars.symboling,
palette=("cubehelix"))
plt.subplot(1,2,2)
plt.title('Symboling vs Price')
sns.boxplot(x=cars.symboling,
y=cars.price, palette=("cubehelix"))
plt.show()
#
plt.figure(figsize=(25, 6))
plt.subplot(1,3,1)
df =
pd.DataFrame(cars.groupby(['CompanyNa
me'])['price'].mean().sort_values(ascendin
g = False))
df.plot.bar(color='green')
plt.title('Company Name vs Average Price')
 plt.subplot(1,3,2)
df =
pd.DataFrame(cars.groupby(['fueltype'])['p
rice'].mean().sort_values(ascending =
False))
df.plot.bar(color='red')
plt.title('Fuel Type vs Average Price')
  plt.subplot(1,3,3)
df =
pd.DataFrame(cars.groupby(['carbody'])['p
rice'].mean().sort_values(ascending =
False))
df.plot.bar(color='blue')
plt.title('Car Type vs Average Price')
plt.show()
plt.figure(figsize=(15,5))
plt.subplot(1,2,1)
plt.title('Door Number Histogram')
sns.countplot(cars.doornumber,
palette=("plasma"))
plt.subplot(1,2,2)
plt.title('Door Number vs Price')
sns.boxplot(x=cars.doornumber,
y=cars.price, palette=("plasma"))
plt.show()
plt.figure(figsize=(15,5))
plt.subplot(1,2,1)
plt.title('Aspiration Histogram')
sns.countplot(cars.aspiration,
palette=("plasma"))
plt.subplot(1,2,2)
plt.title('Aspiration vs Price')
sns.boxplot(x=cars.aspiration,
y=cars.price, palette=("plasma"))
def plot_count(x,fig):
   plt.subplot(4,2,fig)
   plt.title(x+' Histogram')

sns.countplot(cars[x],palette=("magma"))
   plt.subplot(4,2,(fig+1))
   plt.title(x+' vs Price')
   sns.boxplot(x=cars[x], y=cars.price,
palette=("magma"))

plt.figure(figsize=(15,20))
plot_count('enginelocation', 1)
plot_count('cylindernumber', 3)
plot_count('fuelsystem', 5)
plot_count('drivewheel', 7)
plt.tight_layout()
def scatter(x,fig):
   plt.subplot(5,2,fig)
   plt.scatter(cars[x],cars['price'])
   plt.title(x+' vs Price')
   plt.ylabel('Price')
```

```
    plt.xlabel(x)
plt.figure(figsize=(10,20))
scatter('carlength', 1)
scatter('carwidth', 2)
scatter('carheight', 3)
scatter('curbweight', 4)
plt.tight_layout()

def pp(x,y,z):
    sns.pairplot(cars, x_vars=[x,y,z],
y_vars='price',size=4, aspect=1,
kind='scatter')
    plt.show()
pp('enginesize', 'boreratio', 'stroke')
pp('compressionratio', 'horsepower',
'peakrpm')
pp('wheelbase', 'citympg', 'highwaympg')
cars_lr = cars[['price', 'fueltype',
'aspiration','carbody',
'drivewheel','wheelbase',
                'curbweight', 'cylindernumber',
'enginesize', 'boreratio','horsepower',
                 'carlength','carwidth']]
sns.pairplot(cars_lr)
plt.show()
# Defining the map function
def dummies(x,df):
     temp = pd.get_dummies(df[x],
drop_first = True)
    df = pd.concat([df, temp], axis = 1)
    df.drop([x], axis = 1, inplace = True)
    return df
# Applying the function to the cars_lr
cars_lr = dummies('fueltype',cars_lr)
cars_lr = dummies('aspiration',cars_lr)
cars_lr = dummies('carbody',cars_lr)
cars_lr = dummies('drivewheel',cars_lr)
cars_lr =
dummies('cylindernumber',cars_lr)
from sklearn.model_selection import
train_test_split
np.random.seed(0)
df_train, df_test = train_test_split(cars_lr,
train_size = 0.7, test_size = 0.3,
random_state = 100)
from sklearn.model_selection import
train_test_split
np.random.seed(0)
df_train, df_test = train_test_split(cars_lr,
train_size = 0.7, test_size = 0.3,
random_state = 100)
from sklearn.preprocessing import
MinMaxScaler
scaler = MinMaxScaler()
num_vars = ['wheelbase', 'curbweight',
'enginesize', 'boreratio',
'horsepower','carlength','carwidth','price']
df_train[num_vars] =
scaler.fit_transform(df_train[num_vars]
plt.figure(figsize = (30, 25))
sns.heatmap(df_train.corr(), annot = True,
cmap="YlGnBu")
plt.show()
#Dividing data into X and y variables
y_train = df_train.pop('price')
X_train = df_train
#RFE
from sklearn.feature_selection import RFE
from sklearn.linear_model import
LinearRegression
import statsmodels.api as sm
from statsmodels.stats.outliers_influence
import variance_inflation_factor
lm = LinearRegression()
lm.fit(X_train,y_train)
rfe = RFE(lm,n_features_to_select=10)
rfe = rfe.fit(X_train, y_train)
list(zip(X_train.columns,rfe.support_,rfe.ra
nking_))
X_train.columns[rfe.support_]
X_train_rfe =
X_train[X_train.columns[rfe.support_]]
X_train_rfe.head()
```

```
def build_model(X,y):
    X = sm.add_constant(X) #Adding the constant
    lm = sm.OLS(y,X).fit() # fitting the model
    print(lm.summary()) # model summary
    return X
   def checkVIF(X):
    vif = pd.DataFrame()
    vif['Features'] = X.columns
    vif['VIF'] = [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]
    vif['VIF'] = round(vif['VIF'], 2)
    vif = vif.sort_values(by = "VIF", ascending = False)
    return(vif)
X_train_new = build_model(X_train_rfe,y_train)
X_train_new = X_train_rfe.drop(["wheelbase",'carlength', 'wagon'], axis = 1)
X_train_new = build_model(X_train_new,y_train)
#Calculating the Variance Inflation Factor
checkVIF(X_train_new)

lm = sm.OLS(y_train,X_train_new).fit()
y_train_price = lm.predict(X_train_new)
# Plot the histogram of the error terms
fig = plt.figure()
sns.distplot((y_train - y_train_price), bins = 20)
fig.suptitle('Error Terms', fontsize = 20)
# Plot heading
plt.xlabel('Errors', fontsize = 18)
#Scaling the test set
num_vars = ['wheelbase', 'curbweight', 'enginesize', 'boreratio', 'horsepower','carlength','carwidth',"price" ]
df_test[num_vars] = scaler.fit_transform(df_test[num_vars])
#Dividing into X and y
y_test = df_test.pop('price')
X_test = df_test
# Now let's use our model to make predictions.
X_train_new = X_train_new.drop('const',axis=1)
# Creating X_test_new dataframe by dropping variables from X_test
X_test_new = X_test[X_train_new.columns]
# Adding a constant variable
X_test_new = sm.add_constant(X_test_new)
# Making predictions
y_pred = lm.predict(X_test_new)
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
#EVALUATION OF THE MODEL
# Plotting y_test and y_pred to understand the spread.
fig = plt.figure()
plt.scatter(y_test,y_pred)
fig.suptitle('y_test vs y_pred', fontsize=20)
# Plot heading
plt.xlabel('y_test', fontsize=18)
# X-label
plt.ylabel('y_pred', fontsize=16)
print(lm.summary())
#KNN
cd C:\Users\DELL\Desktop\FInal Year Project
import pandas as pd
data=pd.read_csv("CarPrice_Assignment.csv")
data.head()
data.describe()
list(data)
data.dtypes
# import necessary libraries
import pandas as pd
```

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error
from sklearn.inspection import permutation_importance
# load the dataset
auto = pd.read_csv("path/to/automobile.csv")
# select relevant features and split into X and y
X = data[[ 'fueltype', 'aspiration','carbody', 'drivewheel','wheelbase',
                'curbweight', 'cylindernumber', 'enginesize', 'boreratio','horsepower',
                 'carlength','carwidth']]
y = data["price"]
# preprocess the data
X = pd.get_dummies(X, drop_first=True)
X.fillna(X.mean(), inplace=True)
scaler = StandardScaler()
X = scaler.fit_transform(X)
# split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# train the KNN model
knn = KNeighborsRegressor(n_neighbors=5)
knn.fit(X_train, y_train)
# make predictions on the test set
y_pred = knn.predict(X_test)
# evaluate the performance of the model
mse = mean_squared_error(y_test, y_pred)
print("Mean squared error:", mse)
r_squared = knn.score(X_test, y_test)
print("R-squared:", r_squared)
# calculate the feature importances
result = permutation_importance(knn, X_test, y_test, n_repeats=10, random_state=42)
# summarize the feature importances
importances = result.importances_mean
feature_names = pd.get_dummies(data[['fueltype', 'aspiration','carbody', 'drivewheel','wheelbase','curbweight', 'cylindernumber', 'enginesize', 'boreratio','horsepower', 'carlength','carwidth']], drop_first=True).columns
for feature, importance in zip(feature_names, importances):
    print(feature, ":", importance)
```