USED CARS
EXPLORATORY DATA ANALYSIS

SUBMITTED BY:

Adithyan J R
Sreechand S
Malavika S
Hariprasad P

INTRODUCTION

- Problem Definition:
  - There is a huge demand for used cars in the Indian Market today
  - Cars4 U is a budding tech start-up that aims to find foot holes in the business trend of selling pre-owned cars in the market.
- Objective:
  - Explore the dataset and extract insights using Exploratory Data Analysis(EDA).
  - Perform Univariate ,Bivariate ,Multistate Analysis
  - Key Meaningful observation on individual variables and relationship between the variables.
- Data Background
  - The data is based on the sales of used cars and their various attributes manufactured in the past 20 years.
  - https://www.kaggle.com/datasets/sukhmanibedi/cars4u

DATA COLLECTION

Import Necessary libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
# to ignore warnings
import warnings
warnings.filterwarnings('ignore')
```

Read dataset from file using pandas.read_csv()

```
data=pd.read_csv(r"C:\Users\91940\Downloads\used_cars_data.csv")
print(data)
```

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First	26.6 km/kg	998 CC	58.16 bhp	5.0	NaN	1.75
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67 kmpl	1582 CC	126.2 bhp	5.0	NaN	12.50
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	18.2 kmpl	1199 CC	88.7 bhp	5.0	8.61 Lakh	4.50
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77 kmpl	1248 CC	88.76 bhp	7.0	NaN	6.00
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.2 kmpl	1968 CC	140.8 bhp	5.0	NaN	17.74
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
7248	7248	Volkswagen Vento Diesel Trendline	Hyderabad	2011	89411	Diesel	Manual	First	20.54 kmpl	1598 CC	103.6 bhp	5.0	NaN	NaN
7249	7249	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	First	17.21 kmpl	1197 CC	103.6 bhp	5.0	NaN	NaN
7250	7250	Nissan Micra Diesel XV	Kolkata	2012	28000	Diesel	Manual	First	23.08 kmpl	1461 CC	63.1 bhp	5.0	NaN	NaN
7251	7251	Volkswagen Polo GT TSI	Pune	2013	52262	Petrol	Automatic	Third	17.2 kmpl	1197 CC	103.6 bhp	5.0	NaN	NaN
7252	7252	Mercedes-Benz E-Class 2009-2013 E 220 CDI Avan...	Kochi	2014	72443	Diesel	Automatic	First	10.0 kmpl	2148 CC	170 bhp	5.0	NaN	NaN

7253 rows × 14 columns

DATA INFORMATION

Variables	Description
S.NO.	Number of data
Name	Name of the car which includes Brand name and Model name
Location	Location in which car is being sold or available for purchase in cities
Year	Manufacturing year of the car
Kilometers_driven	Total kilometers driven in the car by the previous owner in KM
Fuel_type	The type of fuel used by the car (Petrol,Diesel,CNG,LPG,Electric)
Transmission	The type of transmission used by the car (Automatic,Manual)

Variables	Description
Owner	Type of Ownership (First,Second,Third,Fourth)
Mileage	Standard mileage used by the car company(in kmpl)
Engine	The displacement volume of the engine in CC
Power	Max. power of the engine in bhp
Seats	Number of seats in the car
New_Price	Price of new car of the same model in INR lakhs
Price	Price of the used car (INR lakhs)

Number of rows: 7253

Number of Columns: 14

- Basic Column information

print(data.info())

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7253 entries, 0 to 7252
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   S.No.             7253 non-null   int64  
 1   Name              7253 non-null   object 
 2   Location          7253 non-null   object 
 3   Year              7253 non-null   int64  
 4   Kilometers_Driven  7253 non-null   int64  
 5   Fuel_Type         7253 non-null   object 
 6   Transmission      7253 non-null   object 
 7   Owner_Type        7253 non-null   object 
 8   Mileage           7251 non-null   object 
 9   Engine            7207 non-null   object 
 10  Power             7207 non-null   object 
 11  Seats             7200 non-null   float64
 12  New_Price         1006 non-null   object 
 13  Price             6019 non-null   float64
dtypes: float64(2), int64(3), object(9)
memory usage: 793.4+ KB
```

- Number of unique values in each column

print(data.nunique())

```
S.No.                7253
Name                 2041
Location               11
Year                   23
Kilometers_Driven    3660
Fuel_Type               5
Transmission            2
Owner_Type              4
Mileage               450
Engine                150
Power                 386
Seats                   9
New_Price             625
Price                1373
dtype: int64
```

DATA REDUCTION

Variables to be dropped.

- Serial Number does not have any predictive power.
- New_price has 86 % of Null values.

```
data=data.drop(['S.No.'],axis=1)
data=data.drop(['New_Price'],axis=1)
```

New data information:

```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7253 entries, 0 to 7252
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Name               7253 non-null   object
 1   Location           7253 non-null   object
 2   Year               7253 non-null   int64
 3   Kilometers_Driven  7253 non-null   int64
 4   Fuel_Type          7253 non-null   object
 5   Transmission       7253 non-null   object
 6   Owner_Type         7253 non-null   object
 7   Mileage            7251 non-null   object
 8   Engine             7207 non-null   object
 9   Power              7207 non-null   object
 10  Seats              7200 non-null   float64
 11  Price              6019 non-null   float64
dtypes: float64(2), int64(2), object(8)
memory usage: 680.1+ KB
None
```

DATA CLEANING

- Number of NAN values in each column

print(data.isnull().sum())

```
S.No.                0
Name                 0
Location             0
Year                 0
Kilometers_Driven    0
Fuel_Type            0
Transmission         0
Owner_Type           0
Mileage              2
Engine              46
Power               46
Seats               53
New_Price         6247
Price             1234
dtype: int64
```

- Removing the nan values :

data.dropna()

data.isnull().sum()

```
Out[6]: S.No.                0
        Name                 0
        Location             0
        Year                 0
        Kilometers_Driven    0
        Fuel_Type            0
        Transmission         0
        Owner_Type           0
        Mileage              0
        Engine               0
        Power                0
        Seats                0
        New_Price            0
        Price                0
        dtype: int64
```

DATA CONVERTING

Variables Mileage ,Engine and Power which affects the price of the cars are in “object” datatype. For analysis it is converted to “float64”.

```
7   Mileage            7251 non-null   object
8   Engine             7207 non-null   object
9   Power              7207 non-null   object
```

```
def clean_and_convert(column):
    return pd.to_numeric(column.str.extract('(\d+\.?\d*)')[0], errors='coerce')

data['Mileage'] = clean_and_convert(data['Mileage'])
data['Engine'] = clean_and_convert(data['Engine'])
data['Power'] = clean_and_convert(data['Power'])

data['Mileage']=data['Mileage'].astype(float)
data['Engine'] = data['Engine'].astype(float)
data['Power'] = data['Power'].astype(float)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7253 entries, 0 to 7252
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Name               7253 non-null   object
 1   Location           7253 non-null   object
 2   Year               7253 non-null   int64
 3   Kilometers_Driven  7253 non-null   int64
 4   Fuel_Type          7253 non-null   object
 5   Transmission       7253 non-null   object
 6   Owner_Type         7253 non-null   object
 7   Mileage            7251 non-null   float64
 8   Engine             7207 non-null   float64
 9   Power              7078 non-null   float64
 10  Seats              7200 non-null   float64
 11  Price              6019 non-null   float64
dtypes: float64(5), int64(2), object(5)
memory usage: 680.1+ KB
None
```

FEATURE HANDLING

New Features:-

- Age- Since age of the car is a contributing factor.
- Name column is splitted into two features Brand and Model.

```
from datetime import date
date.today().year
data['Car_Age']=date.today().year-data['Year']

data['Brand']=data.Name.str.split().str.get(0)
data['Model']=data.Name.str.split().str.get(1)+data.Name.str
              .split().str.get(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7253 entries, 0 to 7252
Data columns (total 15 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Name               7253 non-null   object
 1   Location           7253 non-null   object
 2   Year               7253 non-null   int64
 3   Kilometers_Driven  7253 non-null   int64
 4   Fuel_Type          7253 non-null   object
 5   Transmission       7253 non-null   object
 6   Owner_Type         7253 non-null   object
 7   Mileage            7251 non-null   float64
 8   Engine             7207 non-null   float64
 9   Power              7078 non-null   float64
 10  Seats              7200 non-null   float64
 11  Price              6019 non-null   float64
 12  Car_Age            7253 non-null   int64
 13  Brand              7253 non-null   object
 14  Model              7252 non-null   object
dtypes: float64(5), int64(3), object(7)
memory usage: 850.1+ KB
```

Brand Feature has unique values:

```
print(data.Brand.unique())
```

Name: Brand, Length: 7253, dtype: object['Maruti' 'Hyundai' 'Honda' 'Audi' 'Nissan' 'Toyota' 'Volkswagen' ' Tata''Land' 'Mitsubishi' 'Renault' 'Mercedes-Benz' 'BMW' 'Mahindra' 'Ford' 'Porsche' 'Datsun' 'Jaguar' 'Volvo' 'Chevrolet' 'Skoda' 'Mini' 'Fiat' 'Jeep' 'Smart' 'Ambassador' 'Isuzu' 'ISUZU' 'Force' 'Bentley' 'Lamborghini' 'Hindustan' ' 'OpelCorsa']

Brand name ‘Isuzu’ and ‘ISUZU’ are same and they need to be in the uniform order.

‘Mini’ and ‘Land’ needs to be replaces to ‘Mini Cooper’ and ‘Land Rover’ respectively.

```
search_for=['IZUSU','Izusu','Mini','Land']
print(data[data.Brand.str.contains('|'.join(search_for))].head(5))
data["Brand"].replace({"ISUZU": "Isuzu", "Mini": "Mini Cooper","Land":"Land Rover"}, inplace=True)
print(data.Brand.unique())
```

```
['Maruti' 'Hyundai' 'Honda' 'Audi' 'Nissan' 'Toyota' 'Volkswagen' 'Tata'
 'Land Rover' 'Mitsubishi' 'Renault' 'Mercedes-Benz' 'BMW' 'Mahindra'
 'Ford' 'Porsche' 'Datsun' 'Jaguar' 'Volvo' 'Chevrolet' 'Skoda'
 'Mini Cooper' 'Fiat' 'Jeep' 'Smart' 'Ambassador' 'Isuzu' 'Force'
 'Bentley' 'Lamborghini' 'Hindustan' 'OpelCorsa']
```

EDA ANALYSIS

EDA refers to the method of studying and exploring record datasets to indentify

- General patterns in the data.
- Locate outliers
- Identify relationship between variables.

Types of EDA analysis:

- Statistical Analysis
- Univariate Analysis
- Bivariate Analysis
- Multivariate Analysis

➢ STATISTICAL SUMMARY

EDA utilizes precise records to recognize the important tendency, variability, and distribution of variables.
Measures like mean, standard deviation, range, and percentiles are usually used.

	count	mean	std	min	25%	50%	75%	max
Year	7253.0	2013.365366	3.254421	1996.00	2011.00	2014.00	2016.00	2019.00
Kilometers_Driven	7253.0	58699.063146	84427.720583	171.00	34000.00	53416.00	73000.00	6500000.00
Mileage	7251.0	18.141580	4.562197	0.00	15.17	18.16	21.10	33.54
Engine	7207.0	1616.573470	595.285137	72.00	1198.00	1493.00	1968.00	5998.00
Power	7078.0	112.765214	53.493553	34.20	75.00	94.00	138.10	616.00
Seats	7200.0	5.279722	0.811660	0.00	5.00	5.00	5.00	10.00
Price	6019.0	9.479468	11.187917	0.44	3.50	5.64	9.95	160.00
Car_Age	7253.0	10.634634	3.254421	5.00	8.00	10.00	13.00	28.00

Data Description of numerical features

data.describe().T

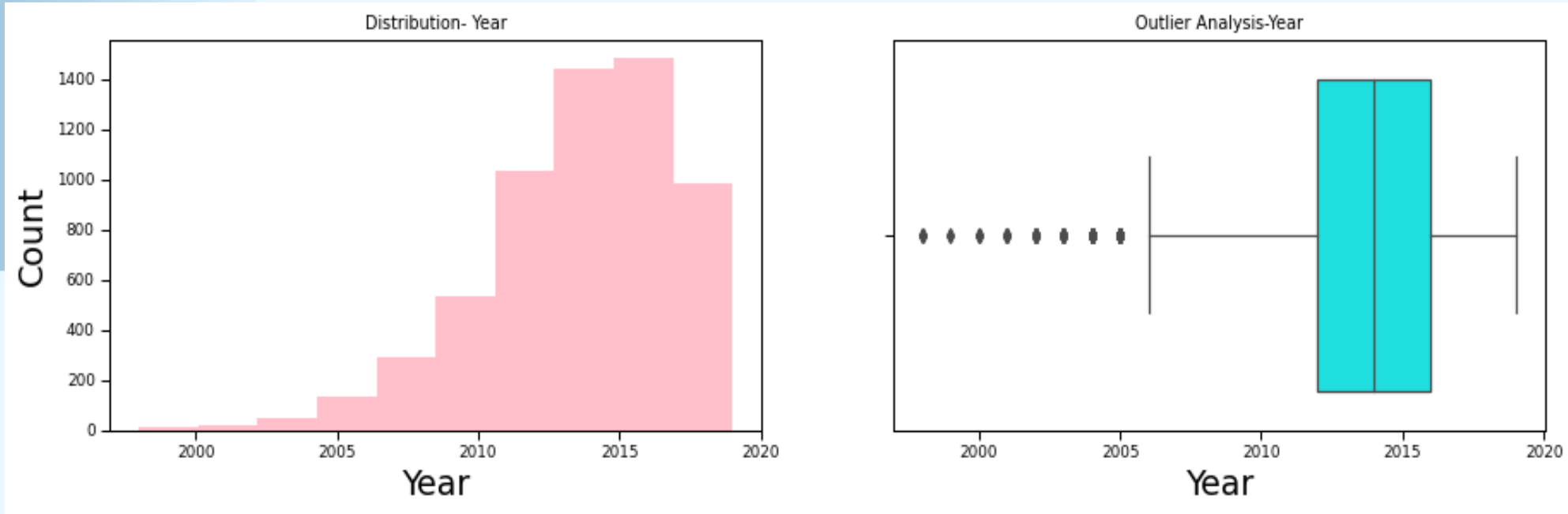
- Year ranges from 1996 to 2019 and has a high range which shows both old model and new model cars are included.
- Kilometers driven has a avg of 58k km that differs highly from max value 6500000km which shows evidence of an outlier.
- Outliers are present in Engine ,Power and in Price.

➢ Univariate Analysis

- Univariate analysis specializes analyzing character variables inside the records set.
- It involves summarizing and visualizing a unmarried variable at a time to understand its distribution, relevant tendency, unfold, and different applicable records.
- **Histogram** is a plot that lets you discover, and show, the underlying frequency distribution (shape) of a set of continuous data. This allows the inspection of the data for its underlying distribution (e.g. normal distribution), outliers, skewness, etc.
- A **Box Plot** is also known as Whisker plot is created to display the summary of the set of data values having properties like minimum, first quartile, median, third quartile and maximum.
- Box Plot is majorly used for viewing the direction of the outliers for features.

➢ UNIVARIATE ANALYSIS cont..

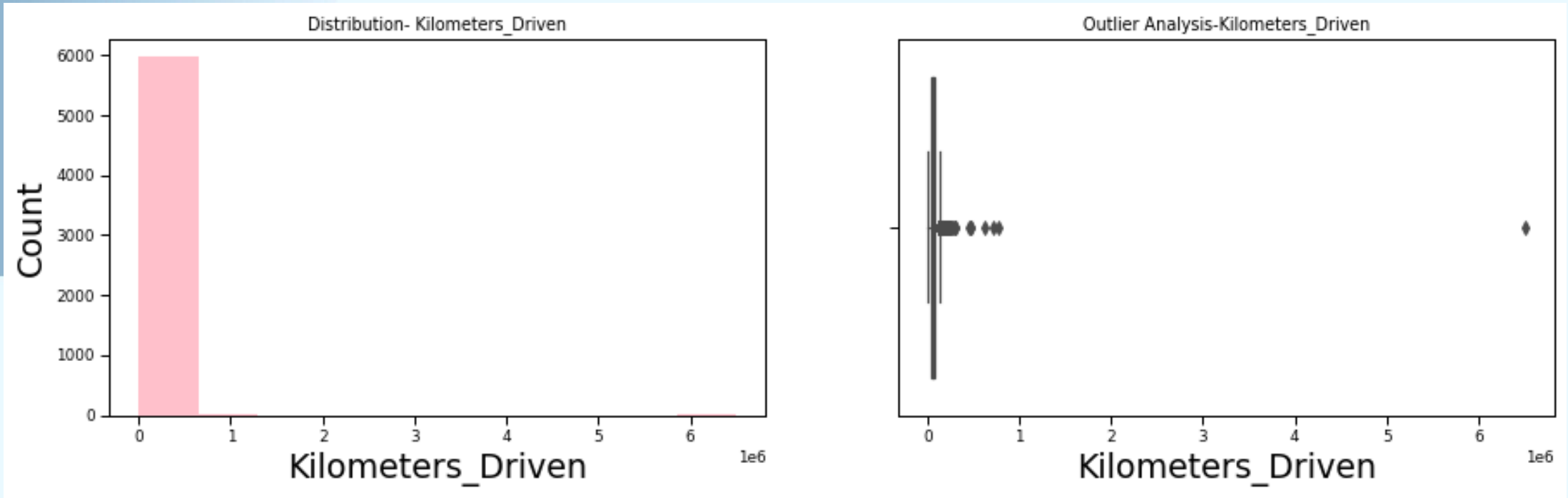
Distribution of Year in which car was manufactured and outlier analysis of Year.



- Skew=-0.84
- Negatively skewed
- Outliers present below 2005

➢ UNIVARIATE ANALYSIS cont..

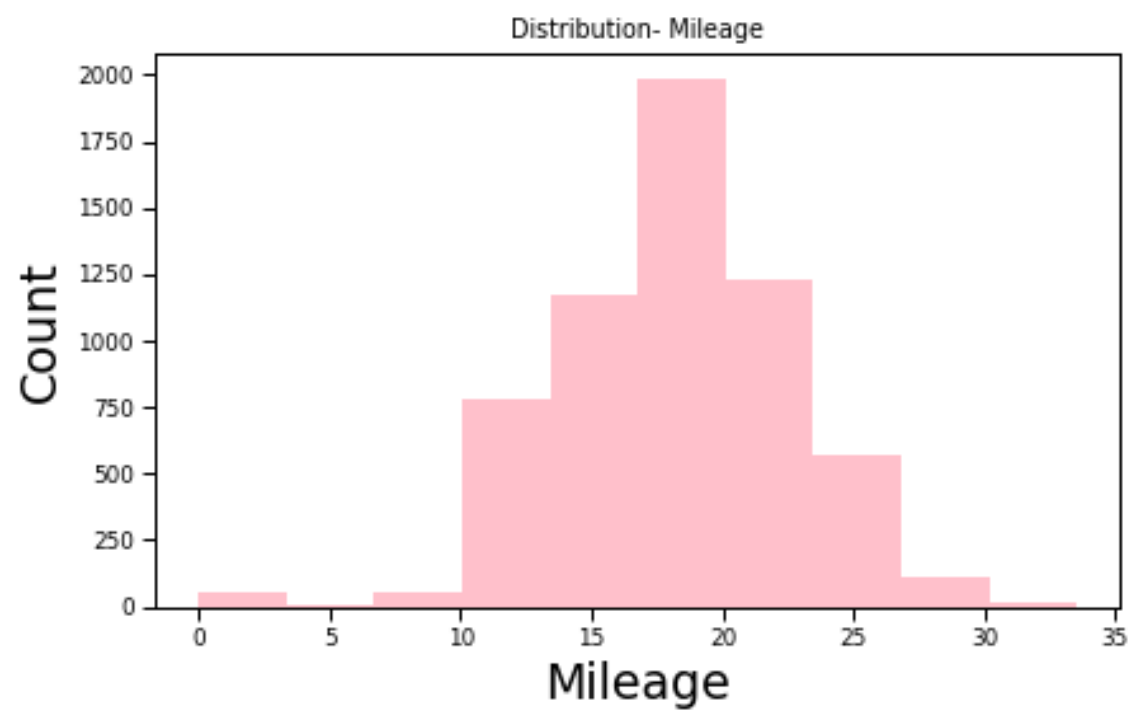
Figure below shows the frequency distribution of Kilometers_driven by the cars in the inventory and outliers in the Kilometers_Driven feature.

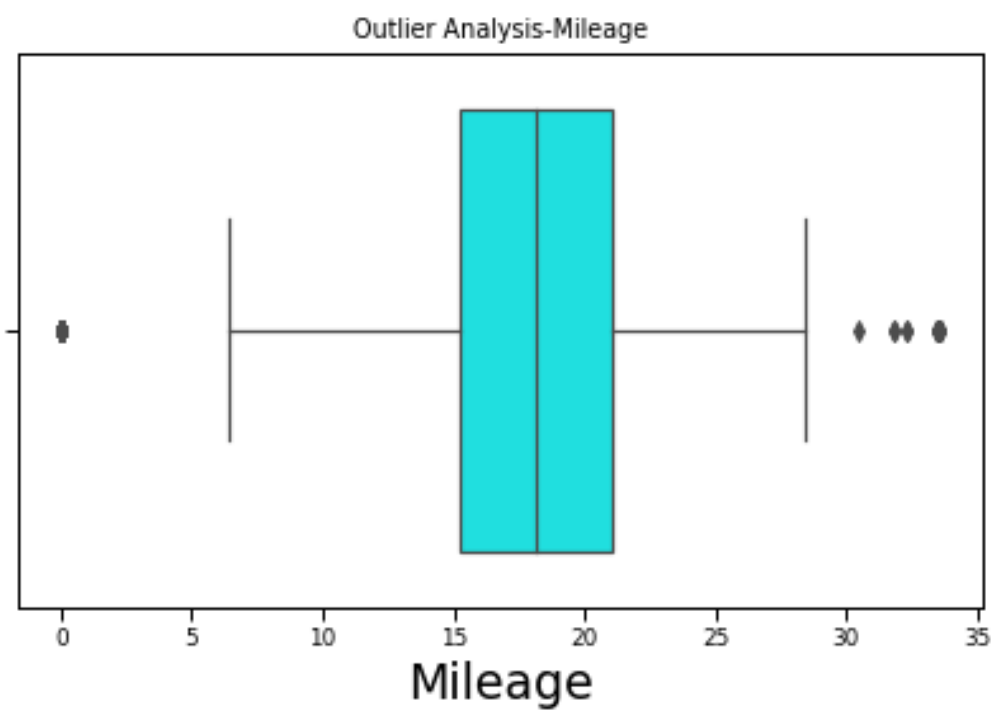


- Skew=61.58
- Have incredibly high range
- Kilometers_driven should be converted to kilometers_driven_log for standardizing
- Major outlier which should be eliminated is 65000000 for BMW X5 xDrive 30d M Sport

➢ UNIVARIATE ANALYSIS cont..

Figure shows the frequency distribution of Mileage and outliers present in the Mileage in the box plot

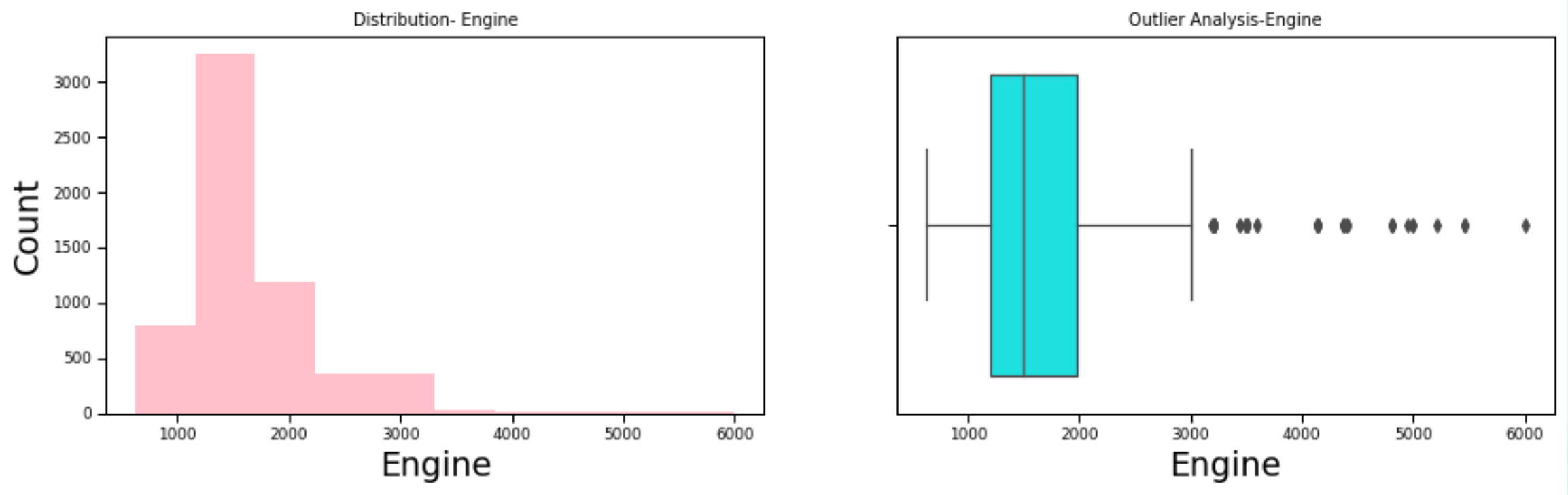




- Skew—0.44
- 81 rows which has value equal to 0 due to data entry error (-ve skewed).
- 18 rows whose mileage is >30 (+ve skewed).

➢ UNIVARIATE ANALYSIS cont..

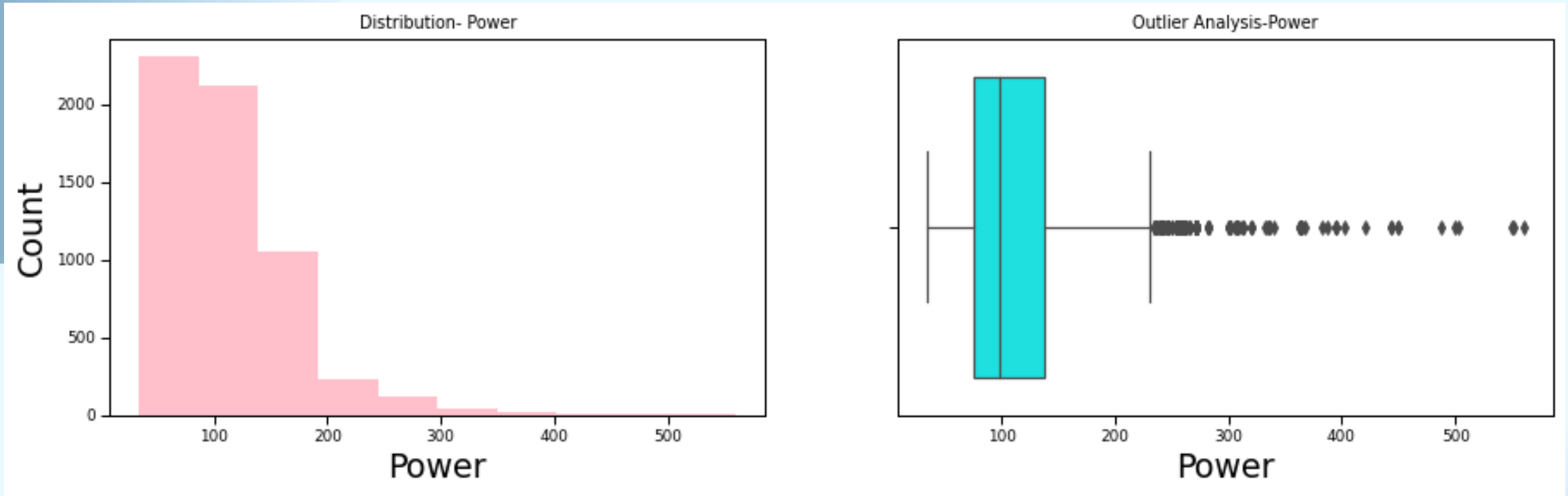
Figure below shows the frequency distribution of Engine of cars available and direction of the outlier in feature Engine



- Skew-1.41
- Outliers are present for engine greater than 3000(65 rows).
- Furthest range of outlier present consists of 6 rows for engine power greater than 5000.

➢ UNIVARIATE ANALYSIS cont..

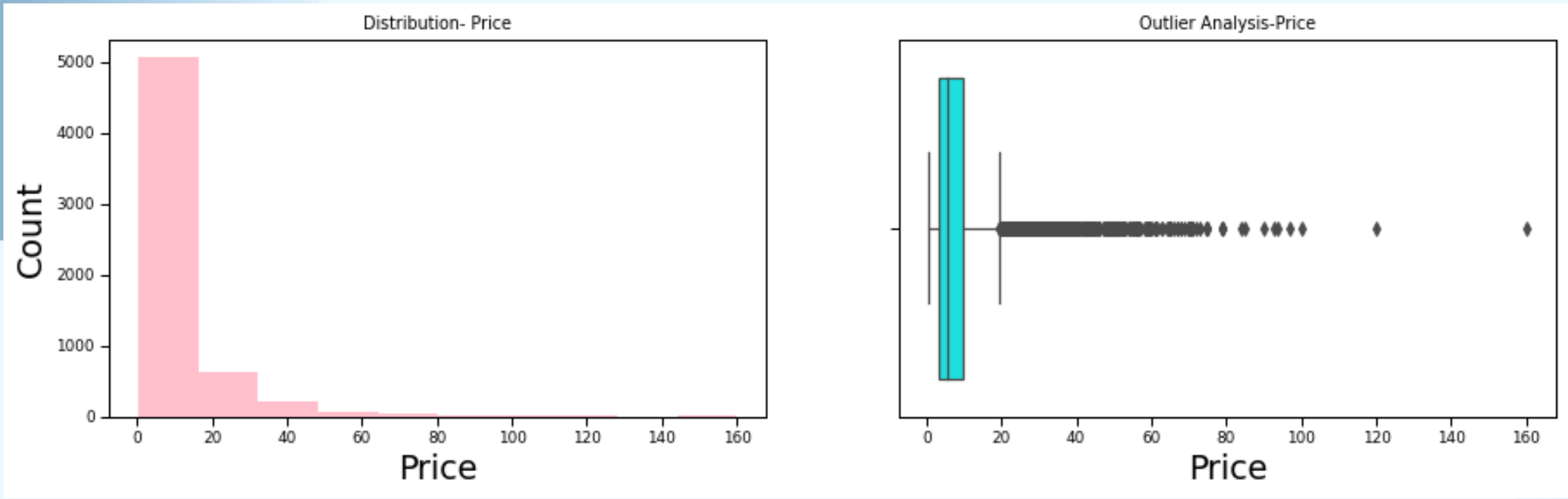
Figure below shows the frequency distribution of power of the cars and outliers in the feature Power



- Skew -1.96
- Outliers ranges from 200bhp to 600bhp
- Max range of outliers that is to be removed consists of 5 rows which is grater than 500bhp.

➢ UNIVARIATE ANALYSIS cont..

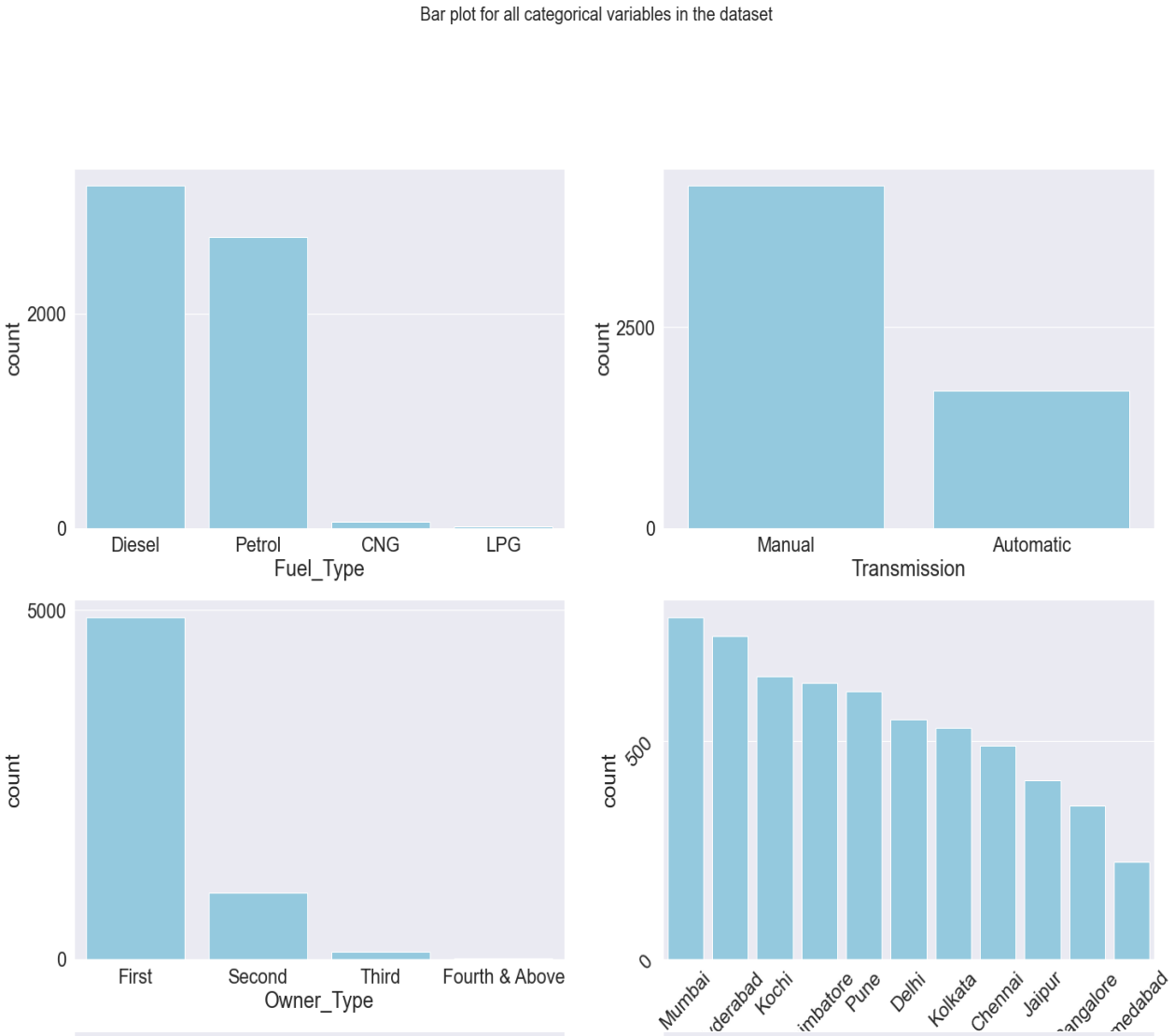
Figure shows the frequency distribution of Age of car and boxplot which indicates where outliers are present in Car_Age Feature



- Skew-3.34
- Outliers are present in 20 lakhs INR to 160 lakh INR.
- Max range of outliers present >100 lakh INR is for Jaguar,Land Rover, Lamborgini.

➢ UNIVARIATE ANALYSIS cont..

Frequency Distribution of categorical columns :

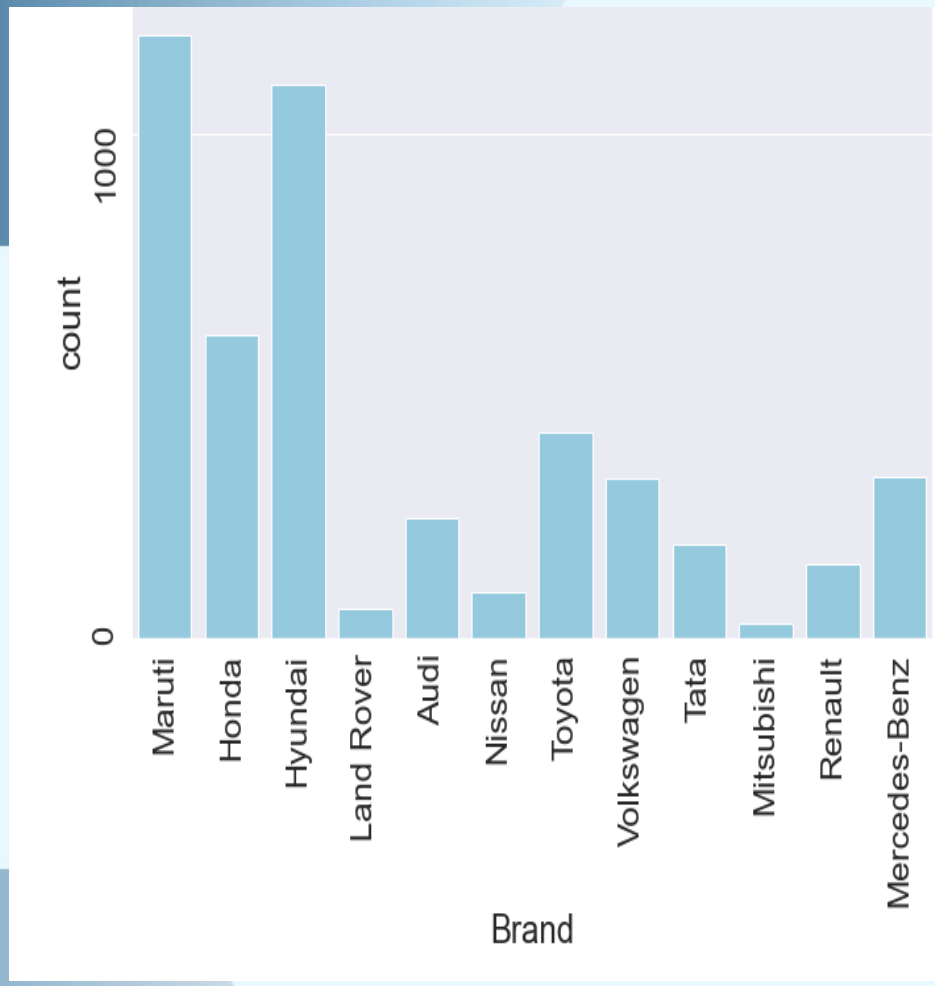


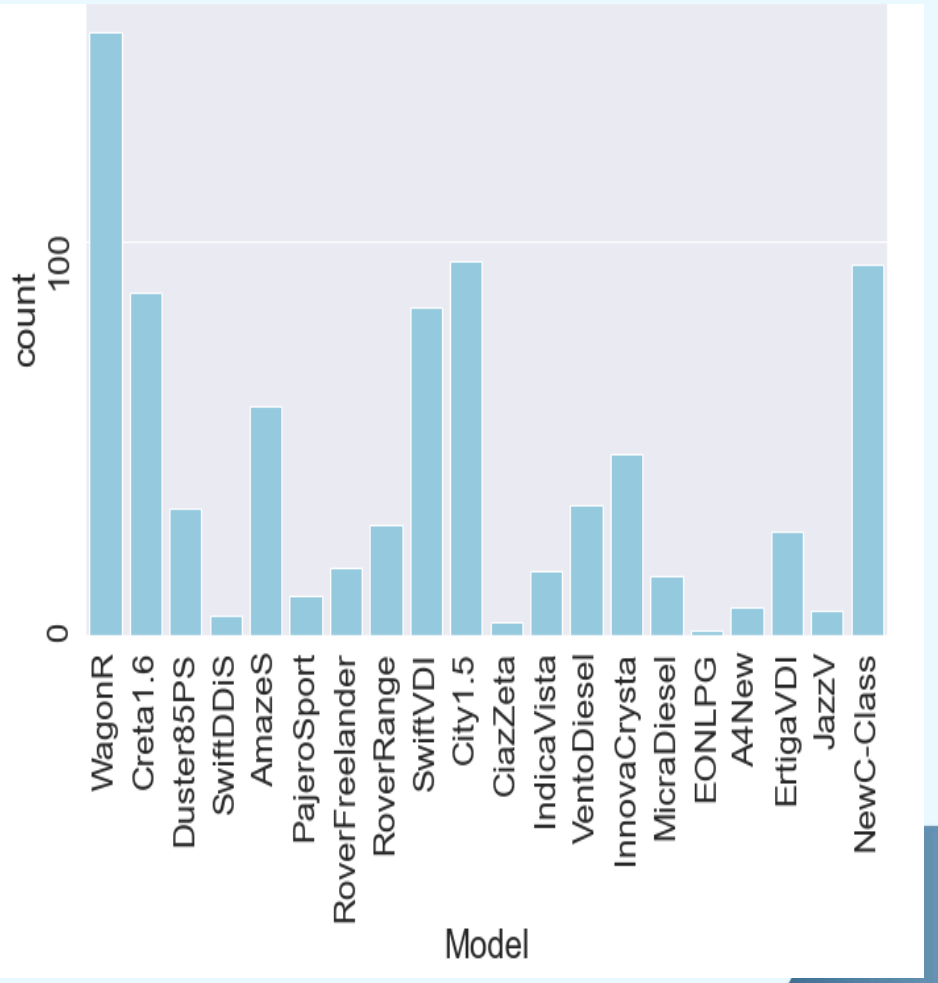
- Mumbai has the highest number of cars available for purchase, followed by Hyderabad and Coimbatore
- ~53% of cars have fuel type as Diesel this shows diesel cars provide higher performance
- ~72% of cars have manual transmission
- ~82 % of cars are First owned cars. This shows most of the buyers prefer to purchase first-owner cars

➢ UNIVARIATE ANALYSIS cont..

Observations of Categorical Columns:

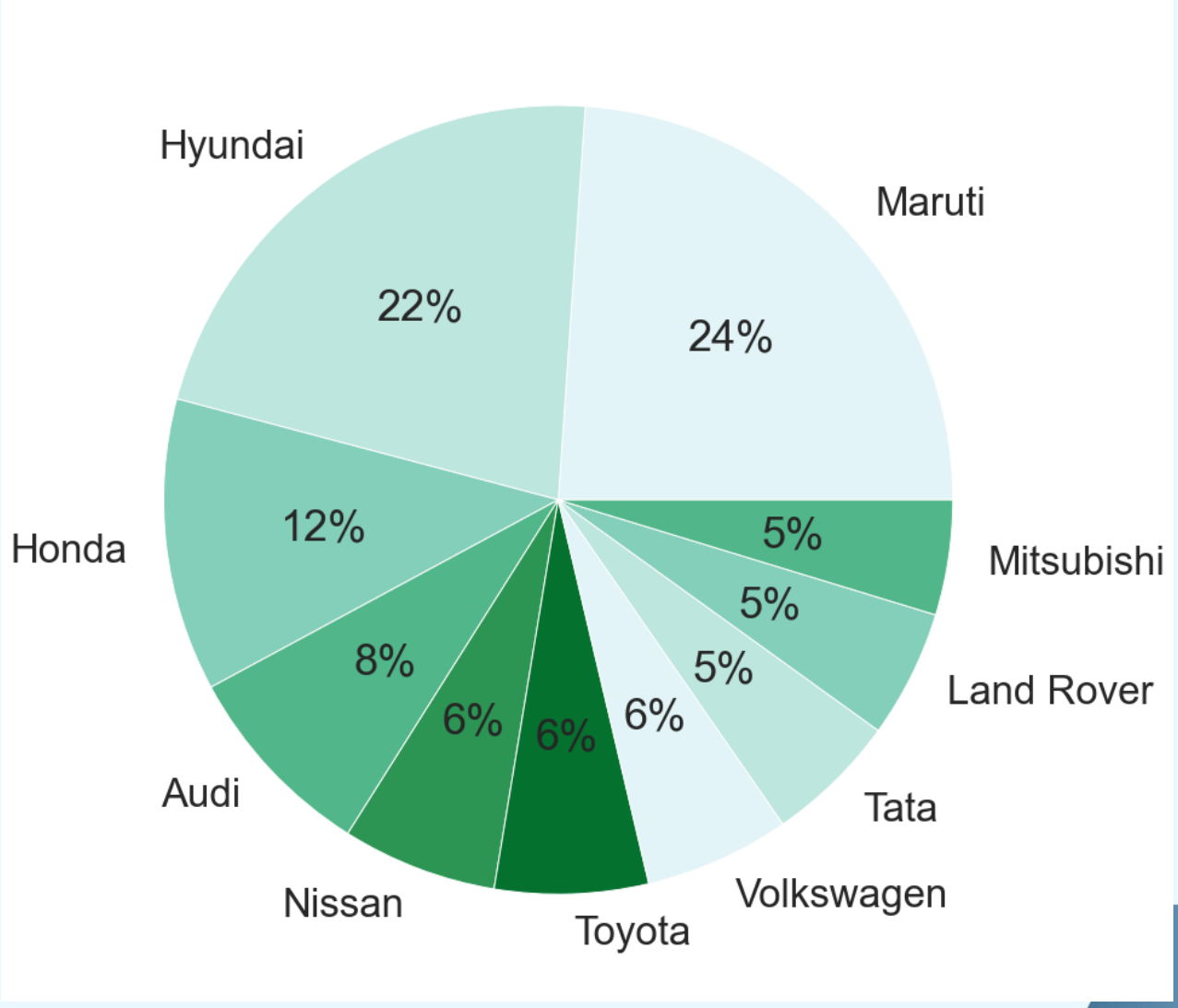
- ~20% of cars belong to the brand Maruti followed by 19% of cars belonging to Hyundai
- WagonR ranks first among all models which are available for purchase





➢ UNIVARIATE ANALYSIS cont..

Figure below show pie plot of percentage distribution of Brand



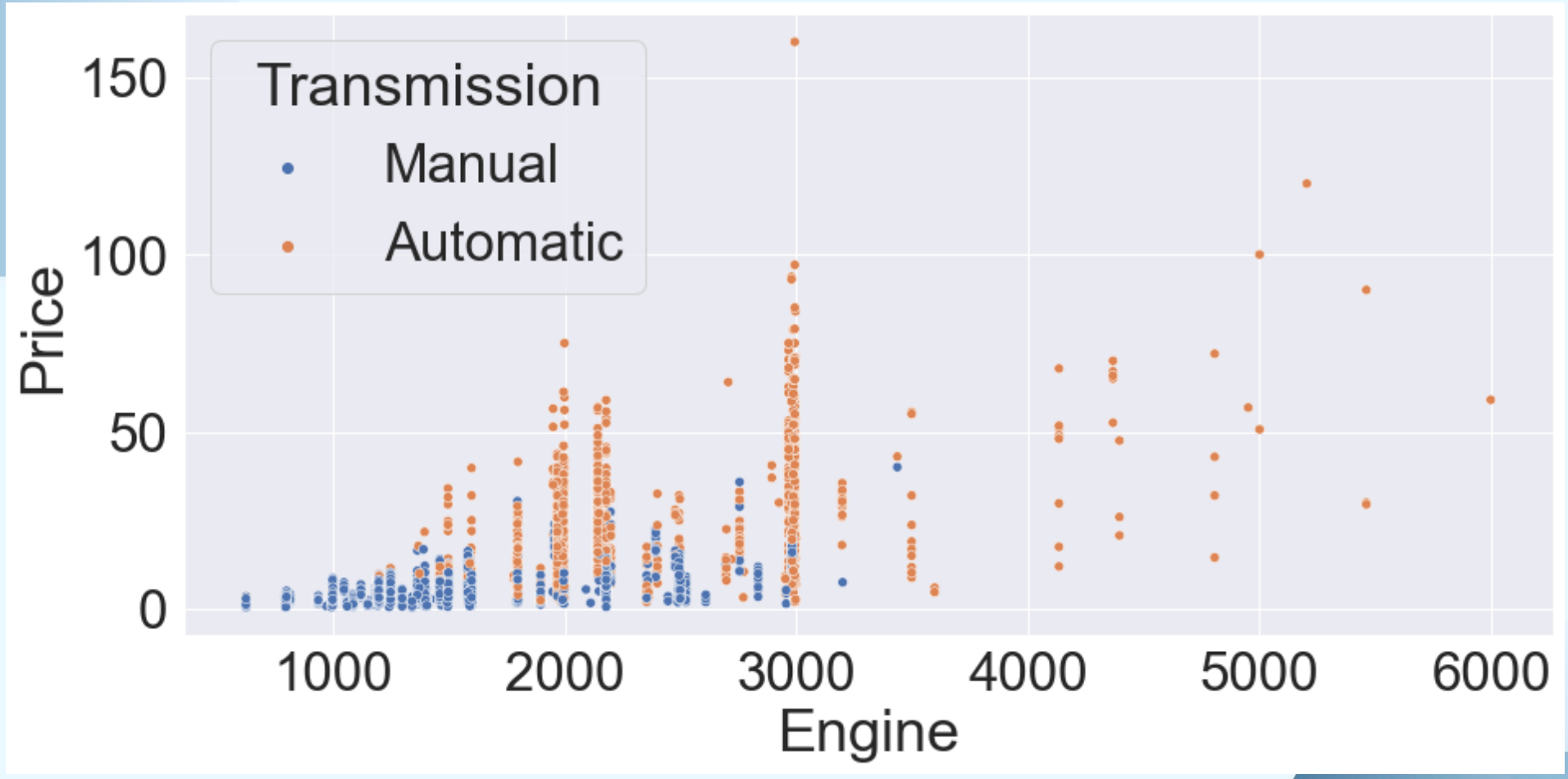
BIVARIATE ANALYSIS

- Bivariate evaluation involves exploring the connection between variables. It enables find associations, correlations, and dependencies between pairs of variables.

- Bivariate Analysis helps to understand how variables are related to each other and the relationship between dependent and independent variables present in the dataset.
- Continous values can be plotted with Pair plots and Scatter plots are widely been used.
- Bar plots can be used to show relationship between categorical features and continuous.

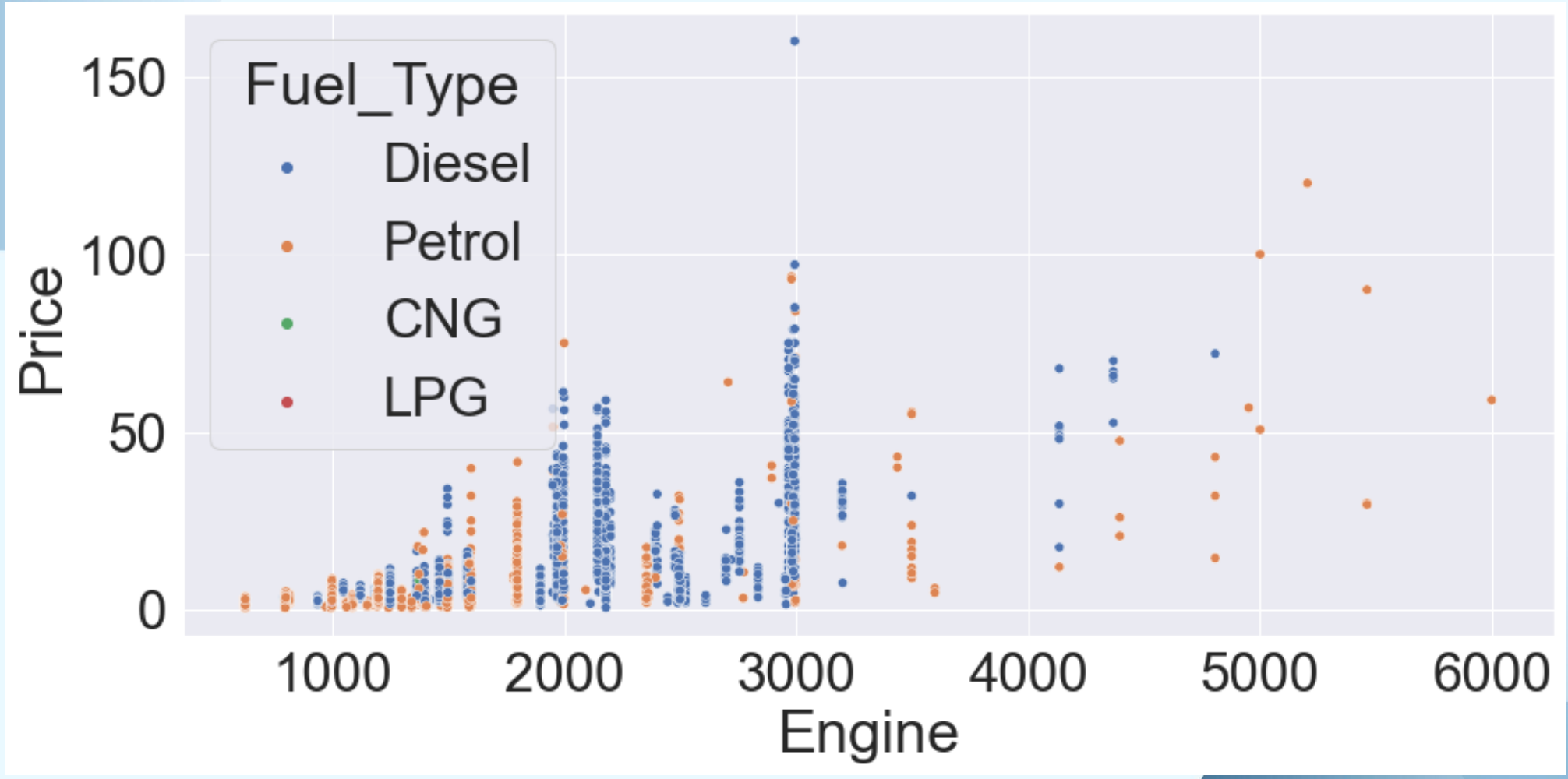
BI-VARIATE ANALYSIS – cont..

Analysis of Price with Engine grouped by transmission.

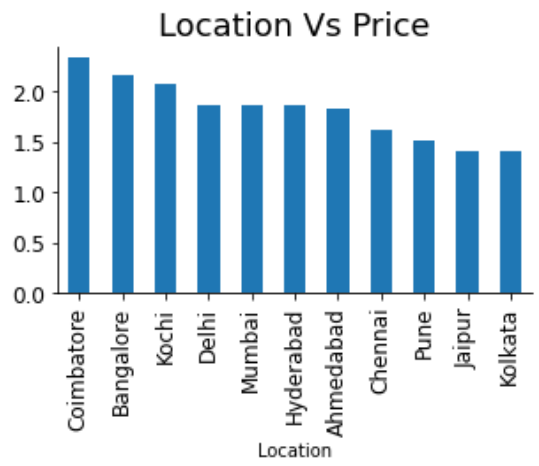


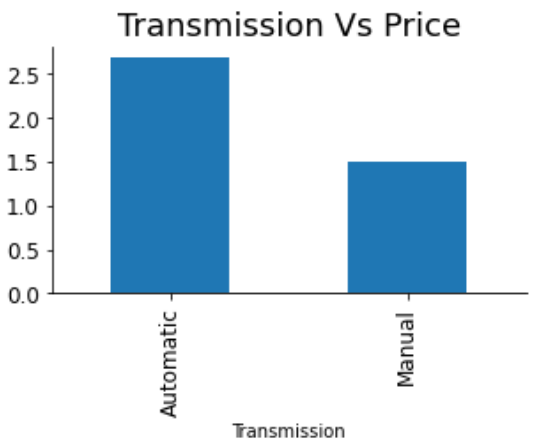
➢ BI-VARIATE ANALYSIS – cont..

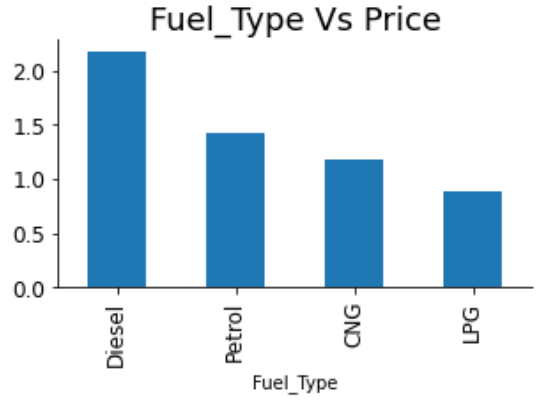
Analysis of Price with Engine grouped by Fuel_Type.

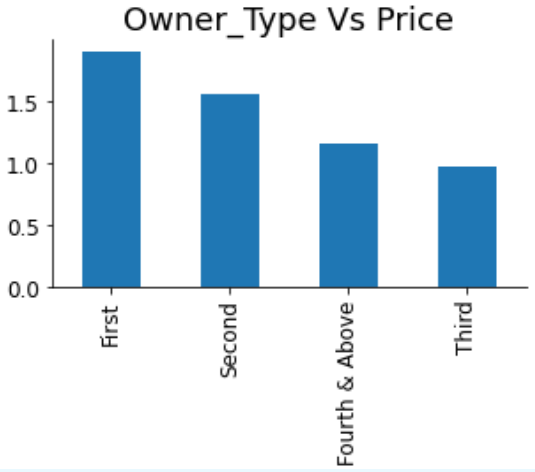


➢ BI-VARIATE ANALYSIS – cont..



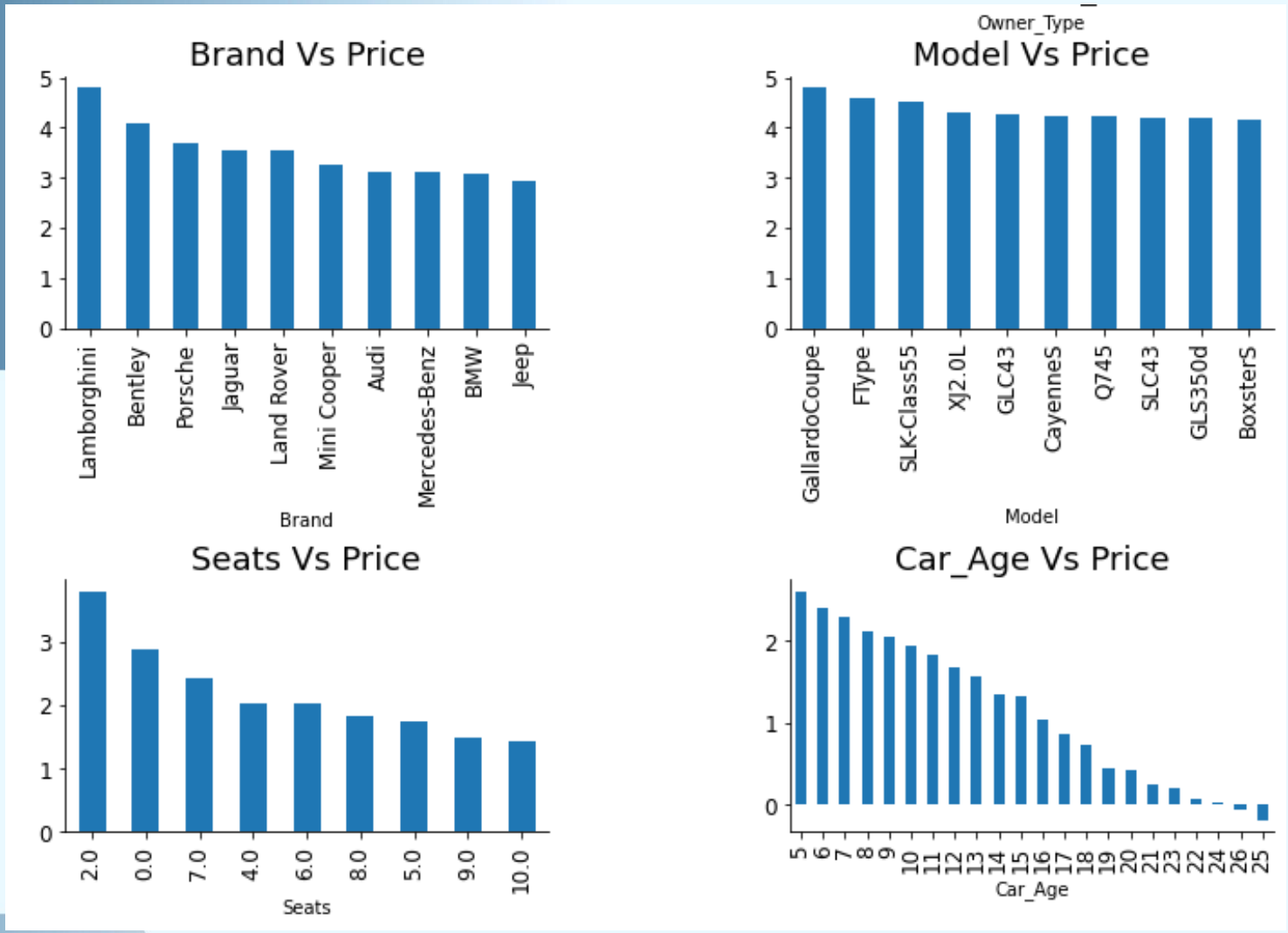






- The price of cars is high in Coimbatore and less price in Kolkata and Jaipur.
- Automatic cars have more price than manual cars.
- Diesel and Electric cars have almost the same price, which is maximum, and LPG cars have the lowest price
- First-owner cars are higher in price, followed by a second
- The third owner's price is lesser than the Fourth and above

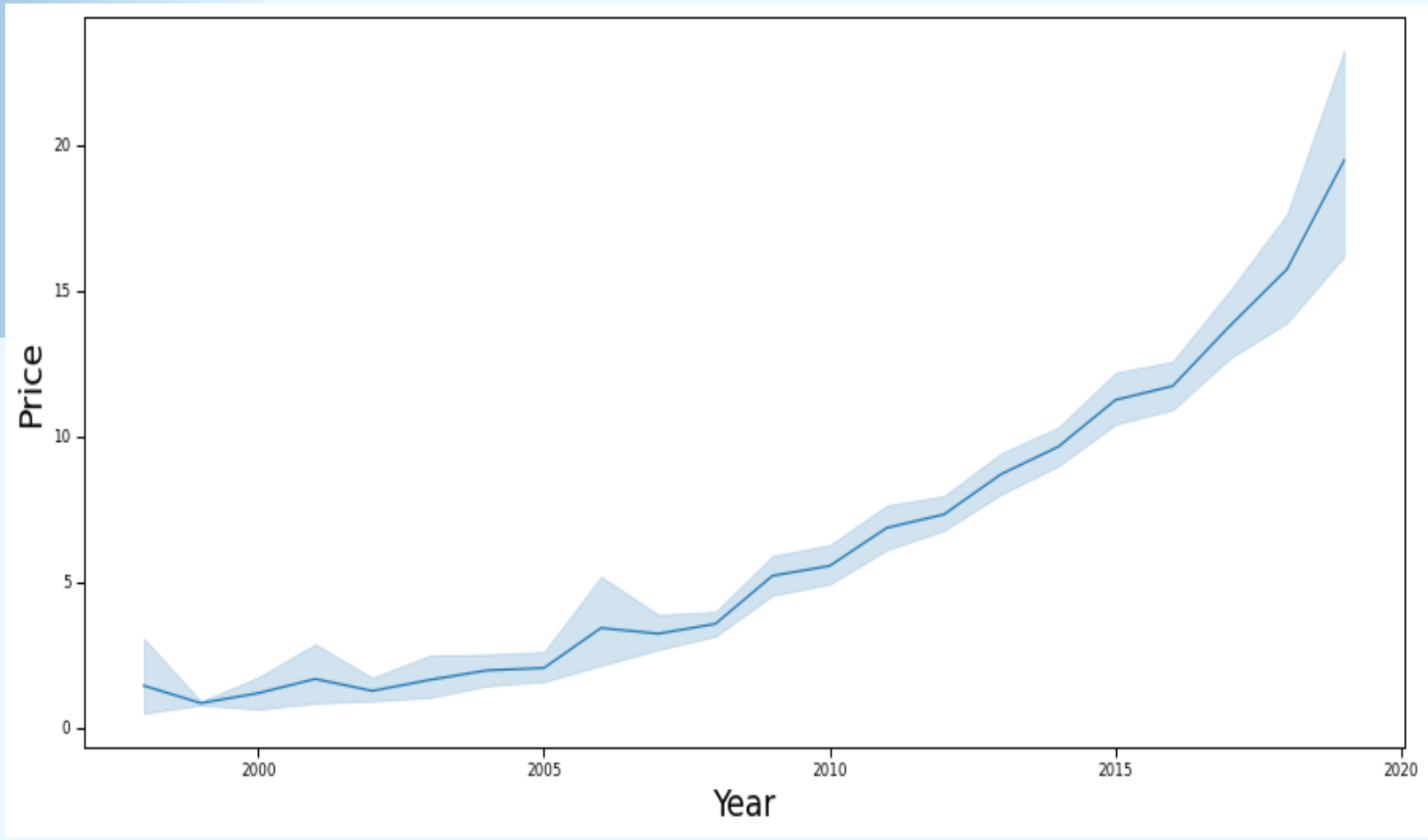
➢ BI-VARIATE ANALYSIS – cont..



- Lamborghini brand is the highest in price
- Gallardocoupe Model is the highest in price
- 2 Seater has the highest price followed by 7 Seater
- The latest model cars are high in price

➢ BI-VARIATE ANALYSIS – cont..

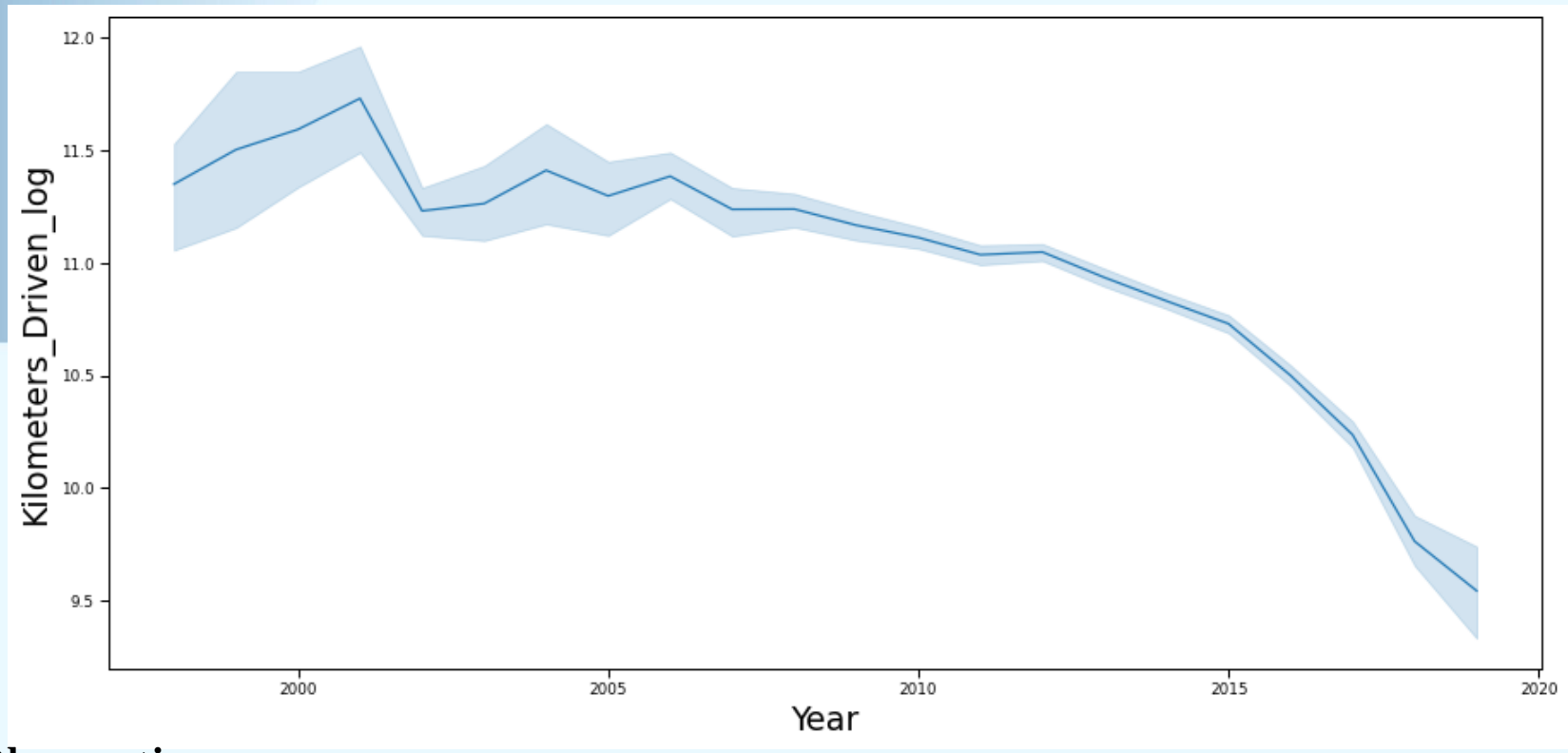
Affect of manufacture year on price of the car



Observation - When manufacture year rises price of the car also increases

➢ BI-VARIATE ANALYSIS – cont..

Relation between Kilometers_driven and year



Observations

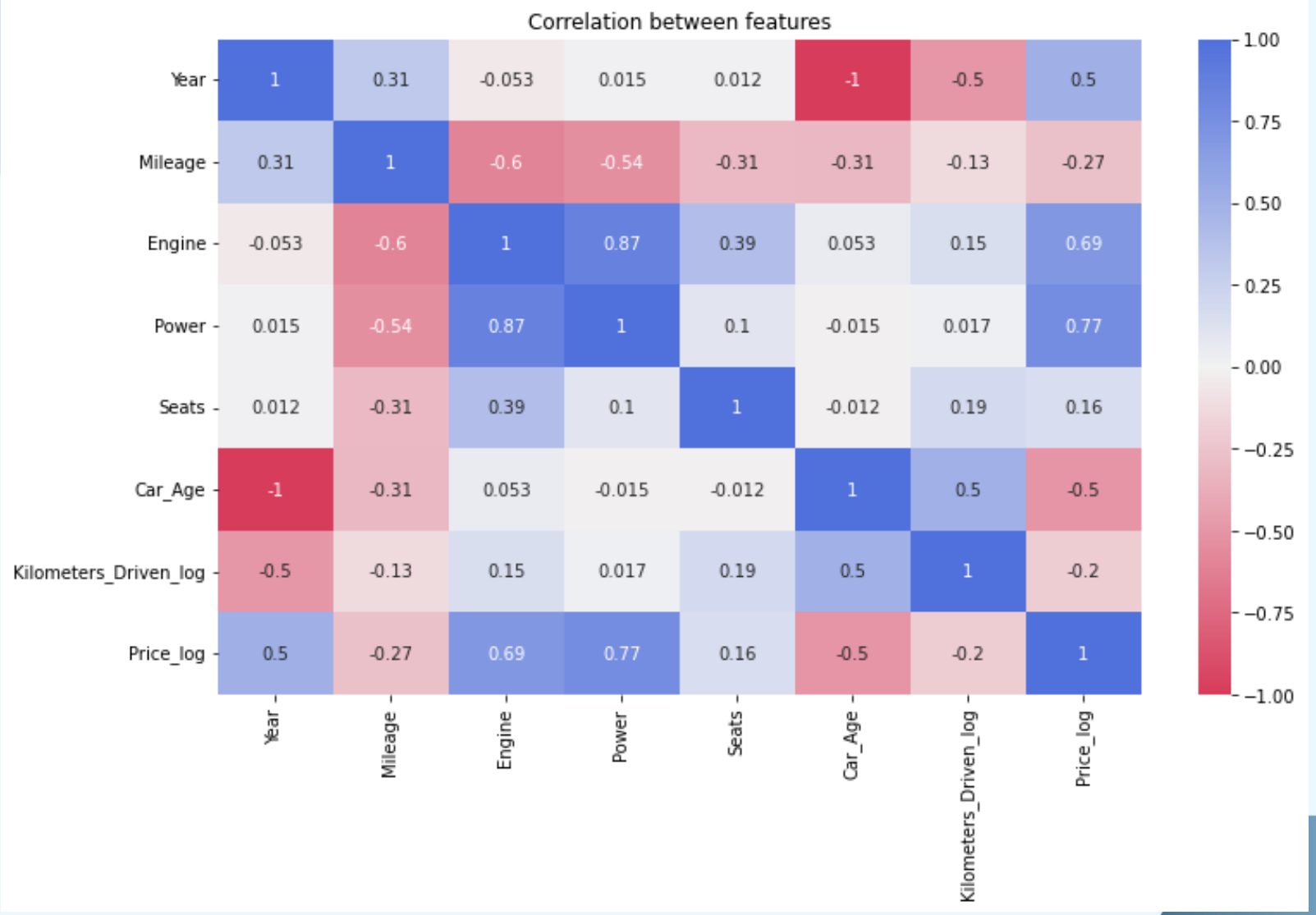
- Year and Kilometers_driven have negative correlation.
- Latest model cars have less usage before being sold.

MULTIVARIATE ANALYSIS

- Multivariate analysis extends bivariate evaluation to encompass greater than two variables.
- It ambitions to apprehend the complex interactions and dependencies among more than one variables in a records set.
- Here correlation heatmap is used. A **correlation heatmap** is a graphical tool that displays the correlation between multiple variables as a color-coded matrix. It's like a color chart that shows us how closely related different variables are.
- In a correlation heatmap, each variable is represented by a row and a column, and the cells show the correlation between them.

➢ MULTIVARIATE ANALYSIS cont..

Figure below shows the Correlation HeatMap between all numerical features.



➢ MULTIVARIATE ANALYSIS cont..

From the Heat map, we can infer the following:

- ❑ The engine has a strong positive correlation to Power 0.86
- ❑ Price has a positive correlation to Engine 0.69 as well Power 0.77
- ❑ Mileage has correlated to Engine, Power, and Price negatively
- ❑ Price is moderately positive in correlation to year.
- ❑ Kilometer driven has a negative correlation to year not much impact on the price
- ❑ Car age has a negative correlation with Price
- ❑ Car Age is positively correlated to Kilometers-Driven as the Age of the car increases; then the kilometer will also increase of car has a negative correlation with Mileage this makes sense

CONCLUSION

- The price of the 2-seat cars is higher than other cars.
- The price of the car decreases as the Age of the car increases.
- Customers prefer to purchase the First owner rather than the Second or Third.
- Automatic Transmission is expensive than Manual.
- Price of the Car increase with higher CC(size) .