

CAR PRICE PREDICTION

Submitted by:

Harneet Kaur rehsi

ACKNOWLEDGMENT

I want to thank my SME Khusboo Garg and Flip Robo Techniques to express our sincere thanks to the following people, without whom we would not have been able to complete this project. I have scrapped the data from Spinny which is an online car dealing website.

INTRODUCTION

Business Problem Framing: With the covid 19 impact in the market, we have seen lot of changes in the car market. Now some cars are in demand hence making them costly and some are not in demand hence cheaper.

Review of Literature: In this project, we need to scrap car-related information i.e: car model, car type, fuel type, price, etc. information. From online websites. There are several websites available regarding car dealing, selling, buying cars or used cars, etc. i.e: Olx, CarDekho, Car24, Spinny and many more. I have scrapped data using one of the Data scraping technique: Selenium. I have scrapped my entire data from the online website name Spinny.

Motivation for the Problem Undertaken: objective behind making this project is due to covid19 there is a huge effect on the market, here I am dealing with changes in the car market their changing price due to covid19. This model will then be used by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand the pricing dynamics of a new market

Analytical Problem Framing


Mathematical/ Analytical Modeling of the Problem: In this project, the dataset contains several rows and columns containing all the necessary information. For removing NaN values present in the dataset we have used several statistical and exploratory data visualization for better understanding and model building for predictions.

Data Sources and their formats: The data contain 74000 rows and 7 columns respectively. containing all the necessary details. Data is scrapped or originated from spinny an online car dealing website.

	Modle Name	Manufacturing Year	Kilometer	Fuel Type	Car Type	Location	Price
0	Maruti Suzuki Wagon R 1.0 LXI	2013	42.5K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹3.05 Lakh
1	Hyundai i10 Era Petrol	2011	84.6K km	Petrol	Manual	Spinny Hub, Ghaziabad (Sahibabad)	₹2.3 Lakh
3	Hyundai Elite i20 Asta 1.2	2016	46.4K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹5.48 Lakh
4	Hyundai New Santro 1.1 Sportz MT	2019	13.5K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹4.72 Lakh
6	Maruti Suzuki Swift VXI	2017	25.9K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹4.99 Lakh
...
7395	Maruti Suzuki Ignis Zeta MT Petrol	2020	11.8K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹5.69 Lakh
7396	Ford EcoSport Titanium 1.5L TDCi	2018	72.1K km	Diesel	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹7.12 Lakh
7397	Kia Seltos HTX D	2020	28.6K km	Diesel	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹14.58 Lakh
7398	Honda Amaze 1.2 VX i-VTEC	2015	34.4K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹4.47 Lakh
7399	Honda City S	2013	51.2K km	Petrol	Manual	Spinny Hub, Garden Galleria Mall, Noida	₹4.98 Lakh

Data Preprocessing Done: Data contains some NaN values. To clean the data I used dropna method and drop all the NaN values present in the particular dataset. Data also contain columns with object-type data which we need to convert into numerical or encode them. So that our machine learning model understands them because the machine learning model understands only numeric type data. I have used replace method are replaced some categorical data with some meaning data by replacing them with some numbers and also used Label Encoder to encode all the categorical or object data into some label so that our machine learning model will understand the information present.

Hardware and Software Requirements and Tools Used: I have used Data Scrapping Technique Selenium for scrapping the data then I have used and imported several libraries like pandas , numpy , matplotlib.pyplot ,seaborn , siciklearning models etc:

 import selenium

```

✚ import pandas as pd
✚ from selenium import webdriver
✚ import warnings
✚ warnings.filterwarnings("ignore")
✚ import numpy as np
✚ import matplotlib.pyplot as plt
✚ %matplotlib inline
✚ import seaborn as sns
✚ from sklearn import metrics
✚ from sklearn.preprocessing import LabelEncoder
✚ from sklearn.linear_model import LogisticRegression
✚ from sklearn.model_selection import cross_val_score
✚ from sklearn.preprocessing import StandardScaler
✚ from sklearn.model_selection import train_test_split
✚ from sklearn.neighbors import KNeighborsClassifier
✚ from sklearn.tree import DecisionTreeClassifier
✚ from sklearn.preprocessing import StandardScaler
✚ from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
✚ from sklearn.ensemble import RandomForestClassifier

```

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods):

```

✚ import numpy as np
✚ import matplotlib.pyplot as plt
✚ %matplotlib inline
✚ import seaborn as sns
✚ from sklearn import metrics
✚ from sklearn.preprocessing import LabelEncoder

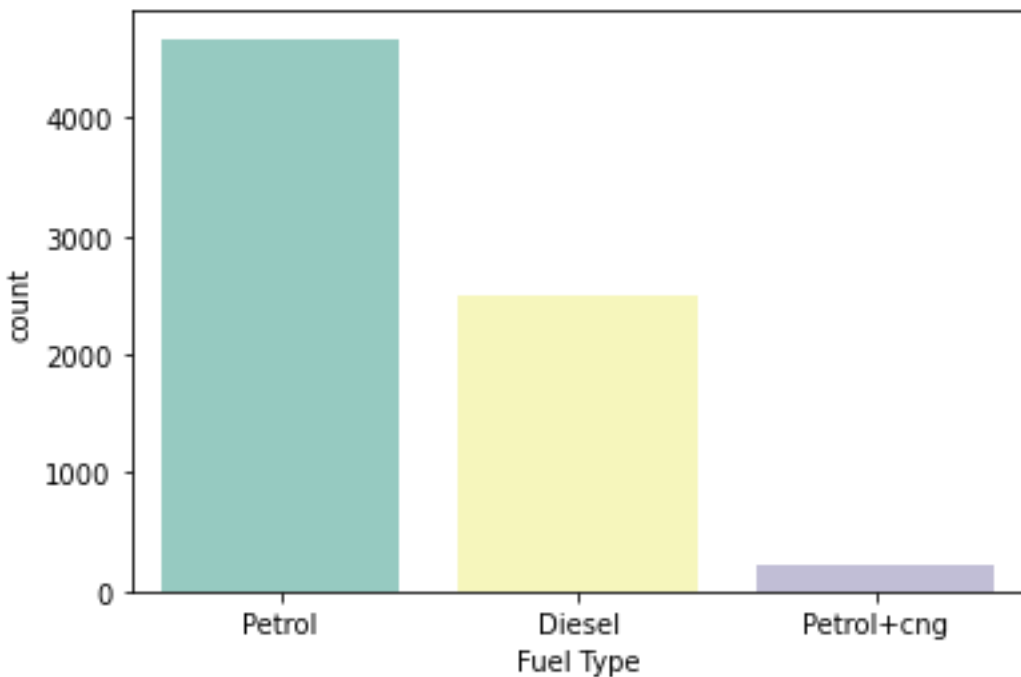
```

These are the approaches I have used for removing NaN values and for Encoding object data into numeric I have used Label Encoder.

Testing of Identified Approaches (Algorithms):

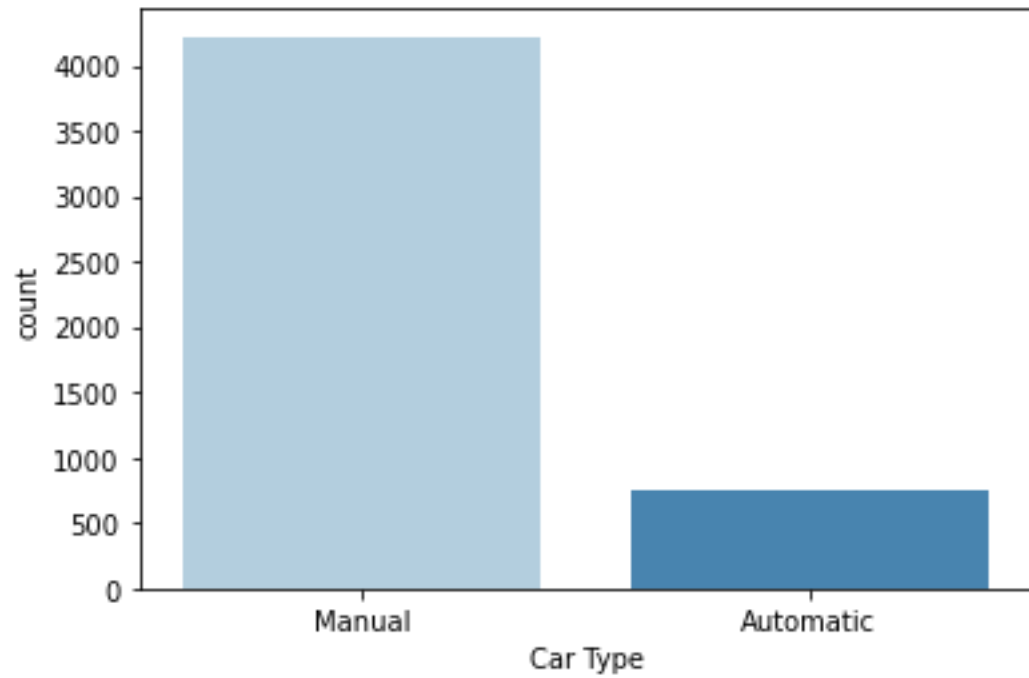
```
from sklearn.preprocessing import StandardScaler
from sklearn import metrics
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.ensemble import RandomForestClassifier
```

Visualizations



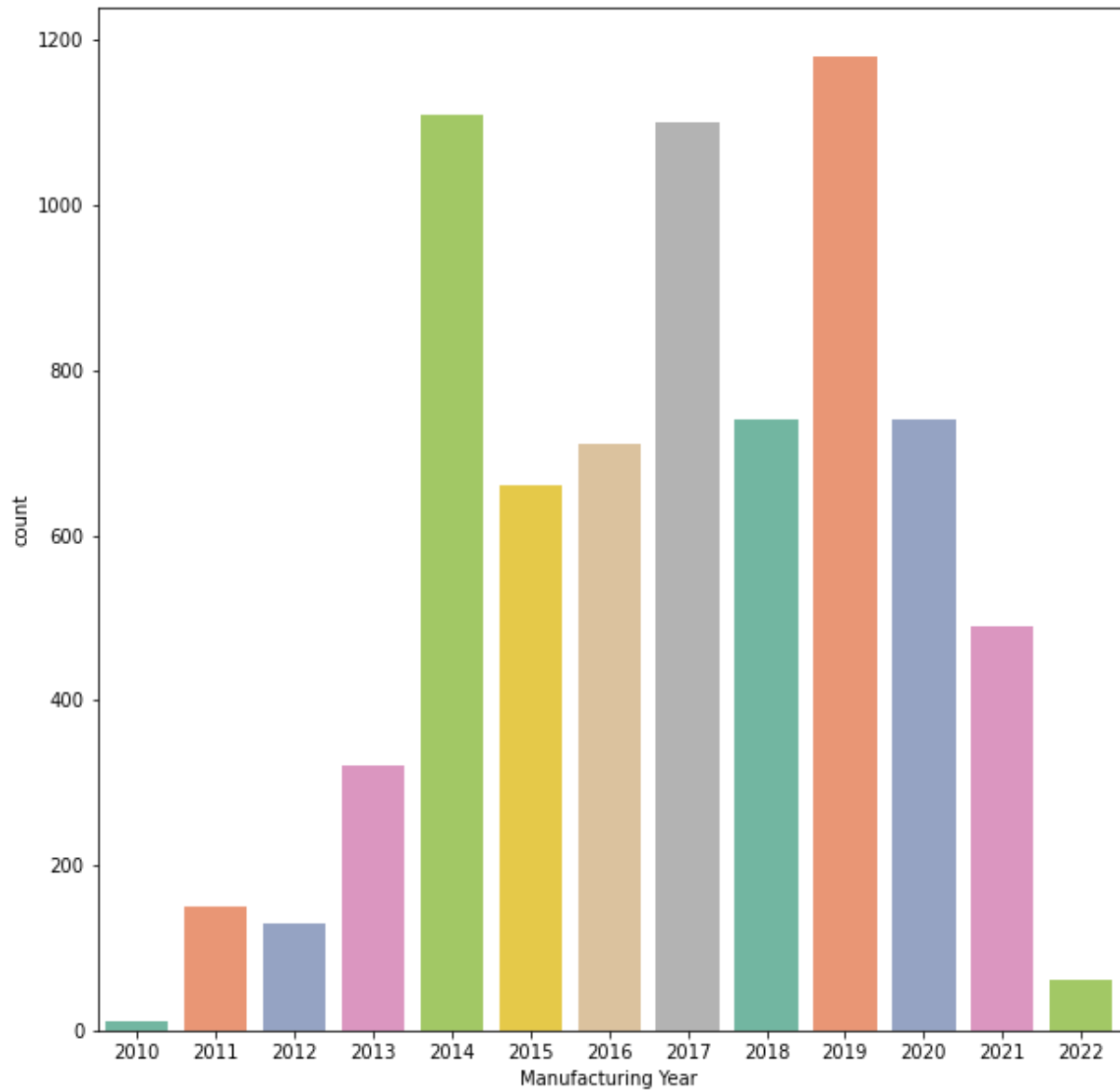
I have used the count plot here for visualizing the Fuel type of the cars

As we can see that there is a huge number of cars having Petrol as their fuel type is sold or bought more. Then Diesel and we can see that Petrol+cng is the least sold or bought fuel type car.



I have used a count-plot here

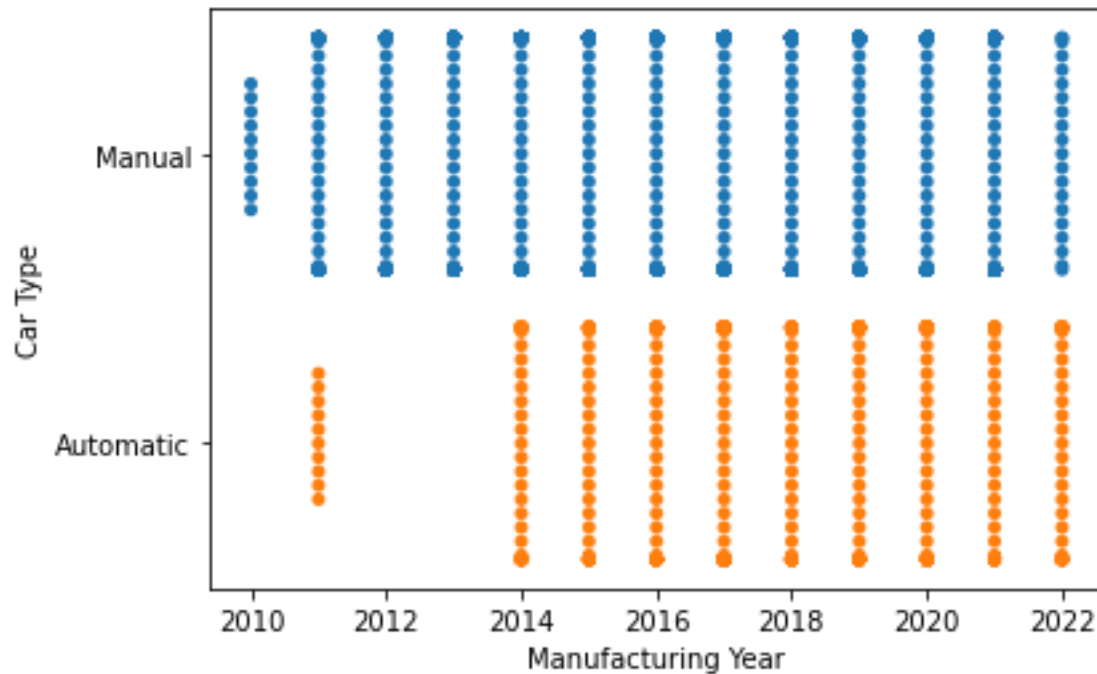
Here it is shown that most bought and sold car type are manual not automatic. People prefer more manual cars as compared to automatic.



This Count-plot represents all the numbers of cars manufactured in that particular year

We can see that the highest car manufacturing year is 2019

And the least manufacturing year is 2010

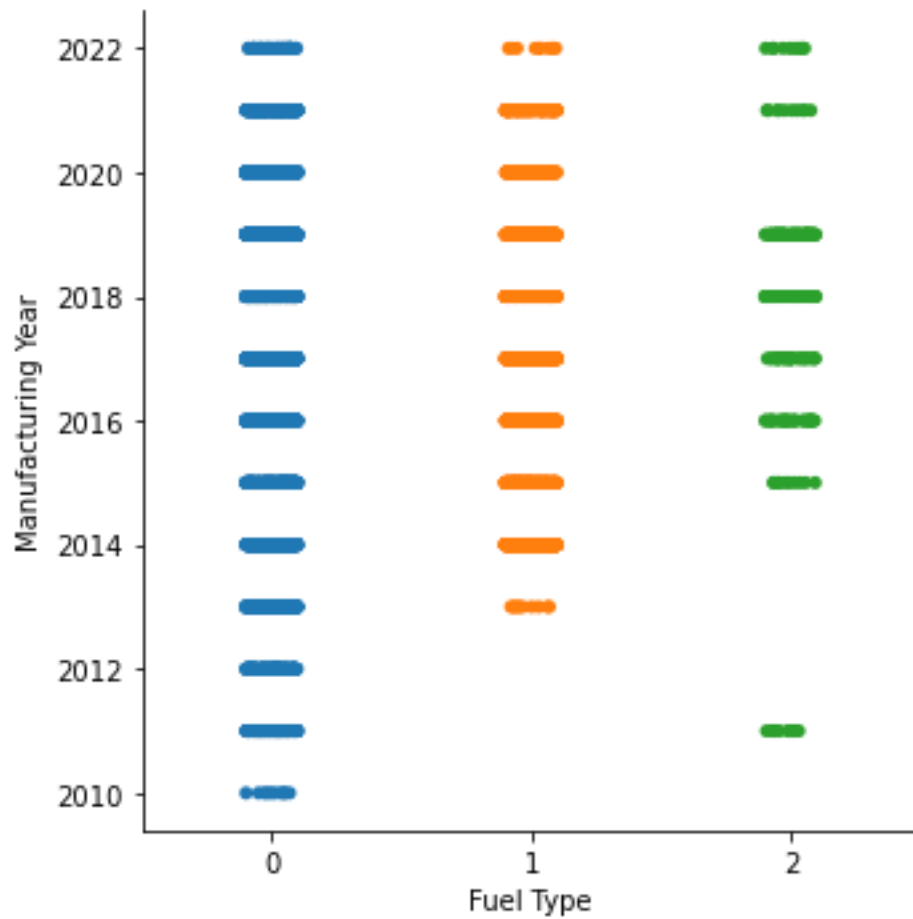


Here I have used a swarm plot for understanding the relationship between manufacturing year and car type

As we can see in 2010 Manual car types were mostly used or build

As the year increases the manufacturing or use of Automatic cars increases

In 2022 both Manual and Automatic are equally manufactured as well as used.

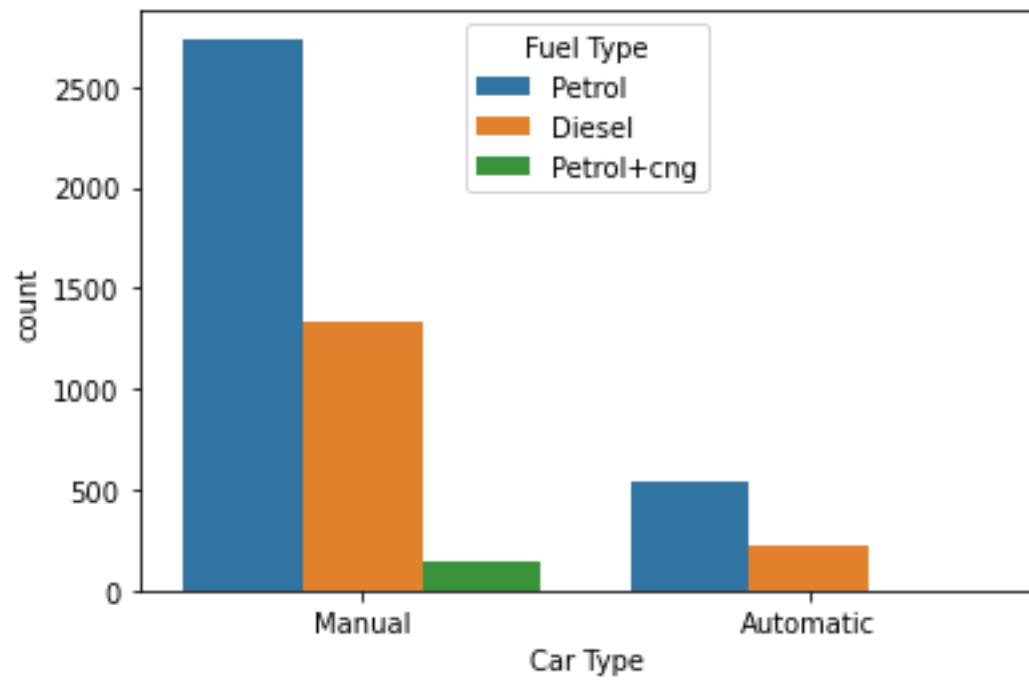


Here I have used a cat plot for understanding the relationship between manufacturing year and Fuel type:

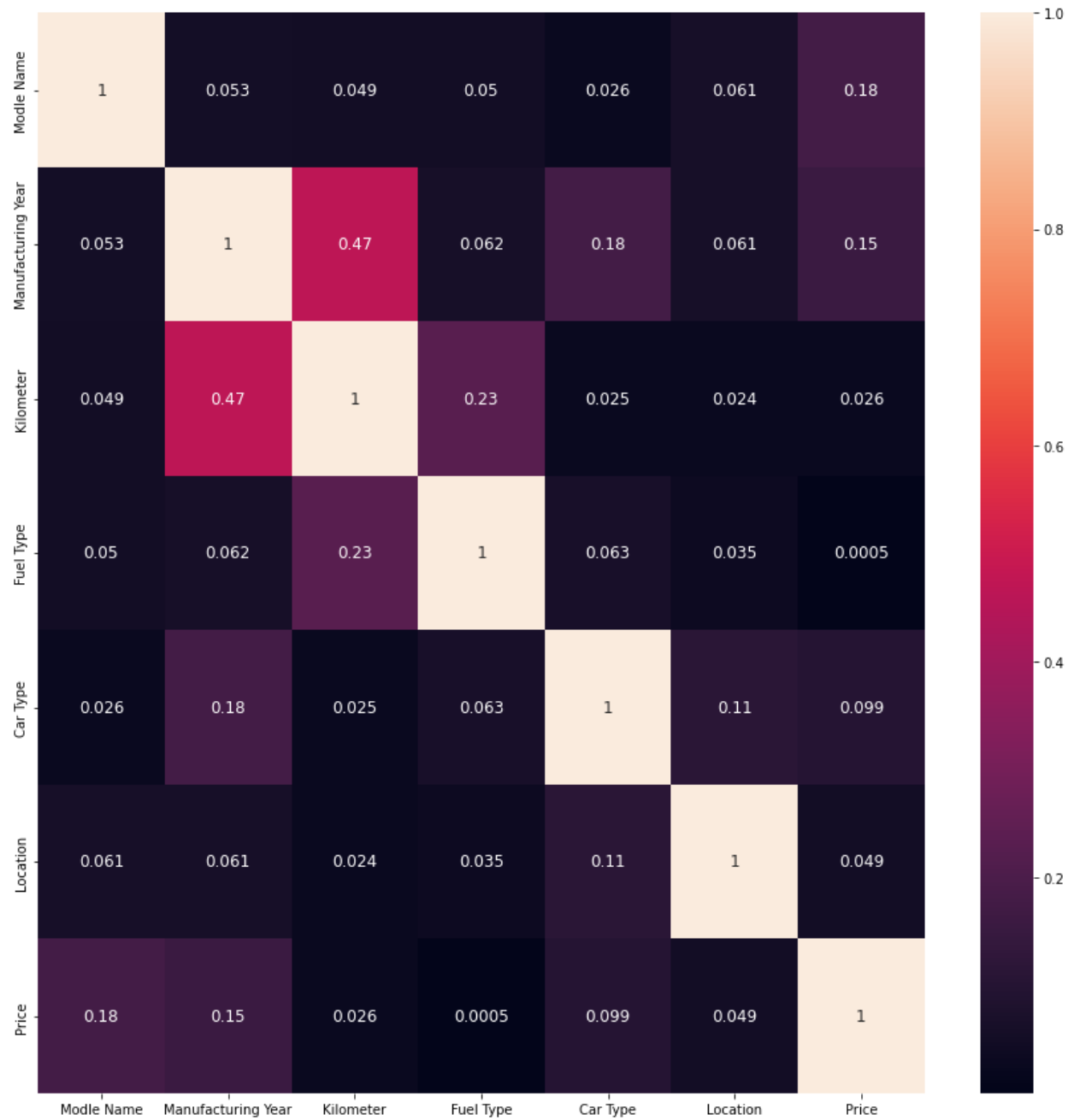
Here: 0 = Petrol

1 = Diesel

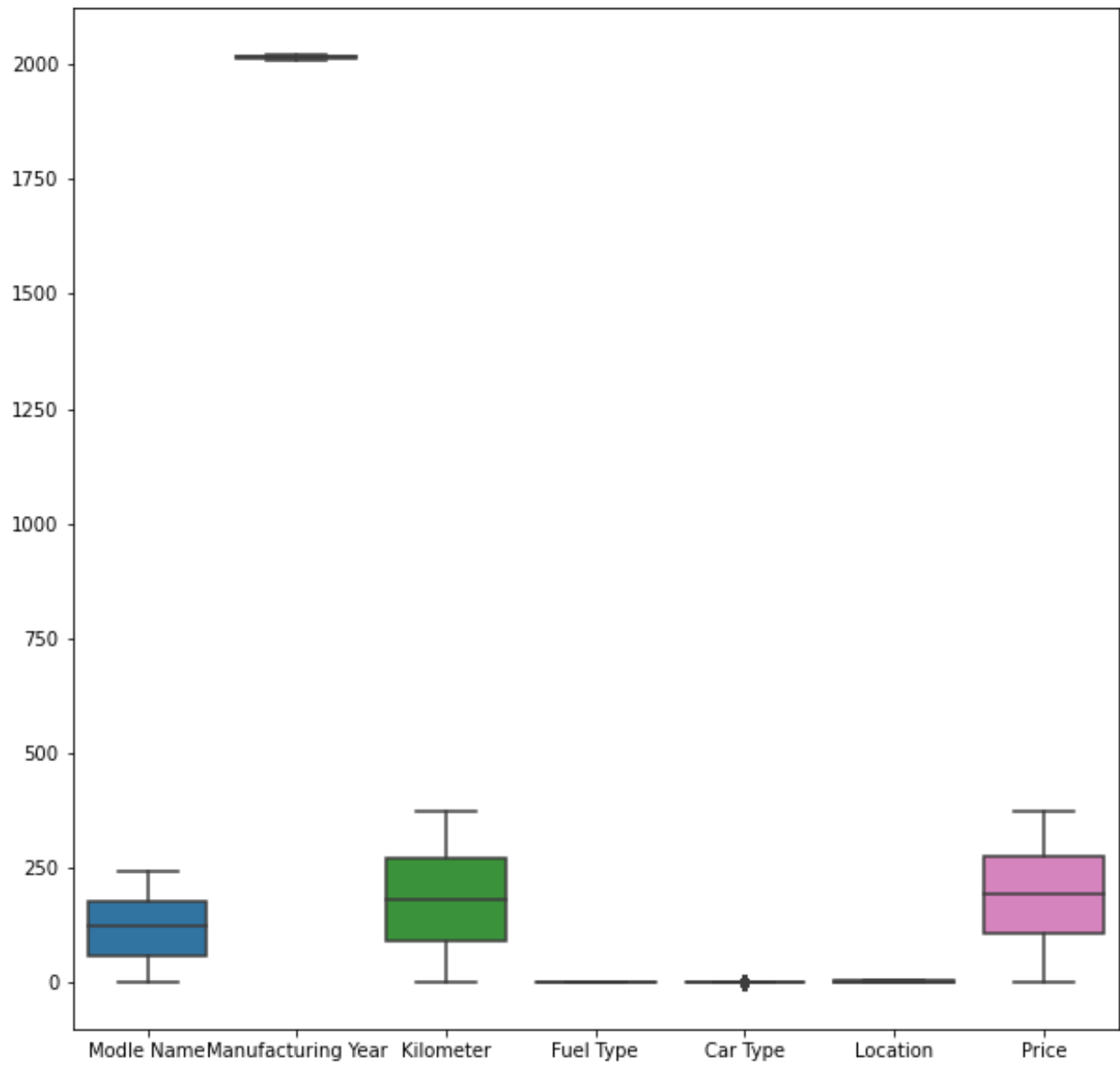
2 = Petrol+cng



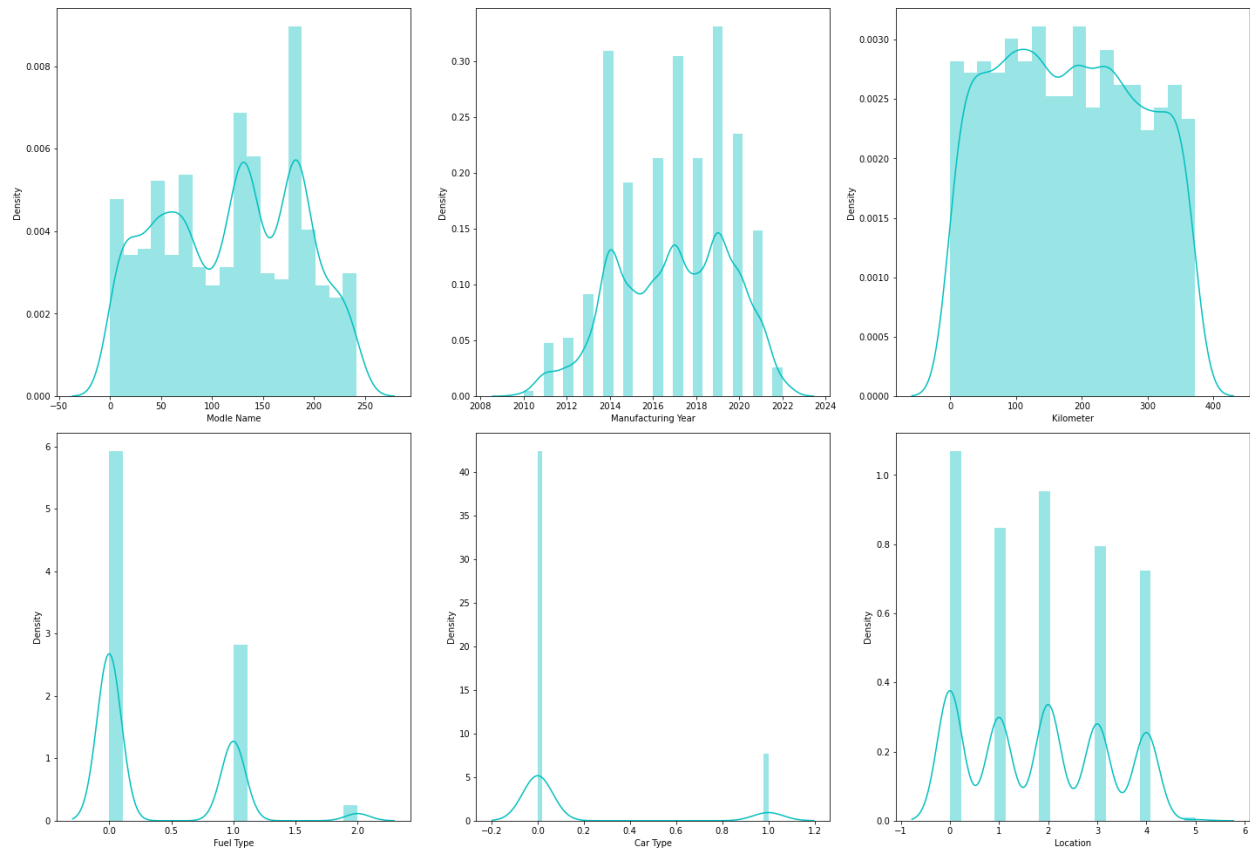
Here I have used Count plot to check the relationship between Car type and Fuel type.



Used heatmap for checking the multicollinearity



Used Boxplot for checking the outliers



Dist-plot of features

Interpretation of the Results: After visualizing the data I have concluded that 2019 is the most manufactured year as the highest number of cars were manufactured in this year and 2010 is the least manufactured year as the lowest number of cars were manufactured in this year. Petrol is the most used fuel type in cars and People prefer more Manual cars as cared to Automatic cars. In 2010 people prefer Manual cars but as the year passed people start to use both Manual and Automatic cars both equally. After using preprocessing methods to convert the data or encode the data and scale the data so that our machine learning model will understand the data properly. The model that I have built “100” accuracy rate. I have tried different machine learning models the accuracy rate is the same in all the models. For checking the model is not overfitted I used cross-validation the result came out that the model is not overfitted.

CONCLUSION

Key Findings and Conclusions of the Study:

After visualizing the data I have concluded that 2019 is the most manufactured year as the highest number of cars were manufactured in this year and 2010 is the least manufactured year as the lowest number of cars were manufactured in this year. Petrol is the most used fuel type in cars and People prefer more Manual cars as compared to Automatic cars. In 2010 people prefer Manual cars but as the year passed people start to use both Manual and Automatic cars both equally. After using preprocessing methods to convert the data or encode the data and scale the data so that our machine learning model will understand the data properly. The model that I have built "100" accuracy rate. I have tried different machine learning models the accuracy rate is the same in all the models. For checking the model is not overfitted I used cross-validation the result came out that the model is not overfitted.

Learning Outcomes of the Study in respect of Data Science

Data contains some NaN values. To clean the data I used the dropna method and drop all the NaN values present in the particular dataset. Data also contain columns with object-type data which we need to convert into numerical or encode them. So that our machine learning model understands them because the machine learning model understands only numeric type data. I have used replace method to replace some categorical data with some meaningful data by replacing them with some numbers and also used Label Encoder to encode all the categorical or object data into some label so that our machine learning model will understand the information present. I have also tried the mode method to replace the data with some meaningful data but it doesn't work here. So I have used dropna method instead