

**CAR PRICE PREDICTION**

Submitted by:

Kunal Chand

**ACKNOWLEDGMENT**

I would like to express my special thanks of gratitude to my teacher

# Dr. Deepika Sharma from Data trained as well as Shubham Yadav(SME) from Flip Robo, who gave me the golden opportunity to do this wonderful project on the topic (CAR PRICE PREDICTION), which also helped me in doing a lot of research and I came to know about so many new things I am thankful to them.Secondly, I would also like to thank my parents who helped me a lot in finalizing this project within the limited time frame.

**INTRODUCTION**

* **Business Problem Framing**

With the covid 19 impact in the market, we have seen a 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. One of our clients works with small traders, who sell used cars. With the change in the market due to covid 19 impact, our client is facing problems with their previous car price valuation machine learning models. So, they are looking for new machine learning models from new data. We have to make a car price valuation model.

* **Conceptual Background of the Domain Problem**

The main problem is found variables that impact most on the label, also the features which are predictable whether the impact od covid on the re-sale market.

* **Review of Literature**

Data exploration is the first step in data analysis and typically involves summarizing the main characteristics of a data set, including its size, accuracy, initial patterns in the data, and other attributes. It is commonly conducted by data analysts using visual analytics tools, but it can also be done in more advanced statistical software, Python. Before it can analyze data collected by multiple data sources and stored in data warehouses, an organization must know how many cases are in a data set, what variables are included, how many missing values there are, and what general hypotheses the data is likely to support. An initial exploration of the data set can help answer these questions by familiarizing analysts with the data with which they are working. We divided the data 8:2 for Training and Testing purposes respectively.

* **Motivation for the Problem Undertaken**

Every problem of Machine learning gives us chance to enhance and develop problem-solving skills. These Problems do’s the same.

When this real-life problem of predicting whether the second-hand car market is impacted by covid-19 or not, whether to rely on old valuation or not is an important aspect to stay in the market and with help of A. I technology we make a completely new model of valuation. As Data scientists it is our role to help companies to understand the market better with newer data, for constructing the valuation model according to that only for make a profitable business.

In this project, I have to scrap used car data from the sites, and for data scrapping, I choose olx autos because of olx autos. which test my patience and intelligence as well.

**Analytical Problem Framing**

* **Mathematical/ Analytical Modelling of the Problem**

As for any basic model building, we have to understand the type of target variable, the data of the target variable is continued or classified.

Data Analysis is always the difficult part, for better understanding different kinds of bar plots, distribution plots are created with the target Column for finding the insights of the dataset we have.

Analytical Modelling always starts with the target variable we have, and in that case, our target variable is Price, for that, we create some distribution plots with the target variable to understand which feature columns help to learn the model best and which feature columns reduce the accuracy of the model.

And after finding the relation and correlation with the target variable we choose either Regression Model or Classification Model. Here in this problem, our target feature column is continuous so we build our Machine Learning model on Regression.

* **Data Sources and their formats**

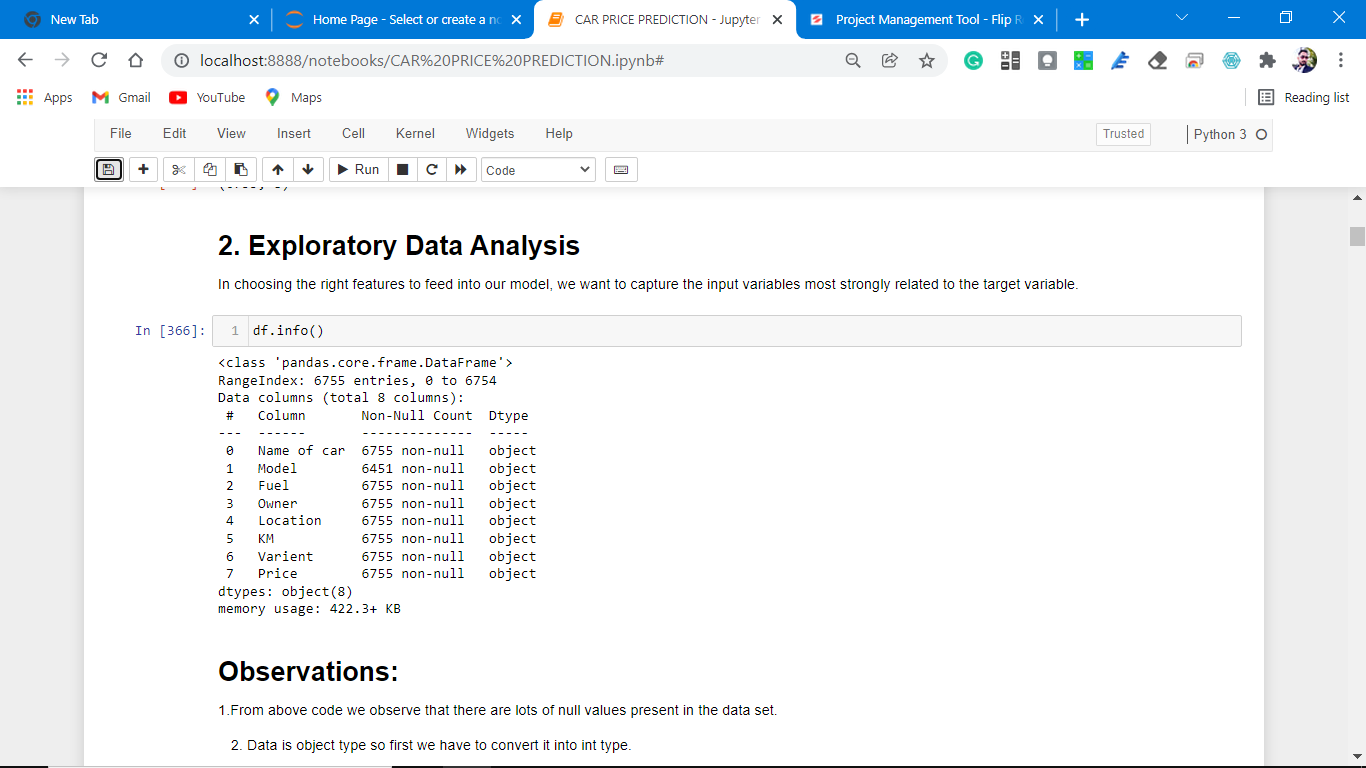
As in this project, we have to scrap a minimum of 5000 used cars data from different available used car sites present at different websites, and for that, we use python and selenium to scrap data from olx autos and save them in excel format.

After loading the data in python and making a frame our dataset looks like this.

**Dataset looks as follows-**

****

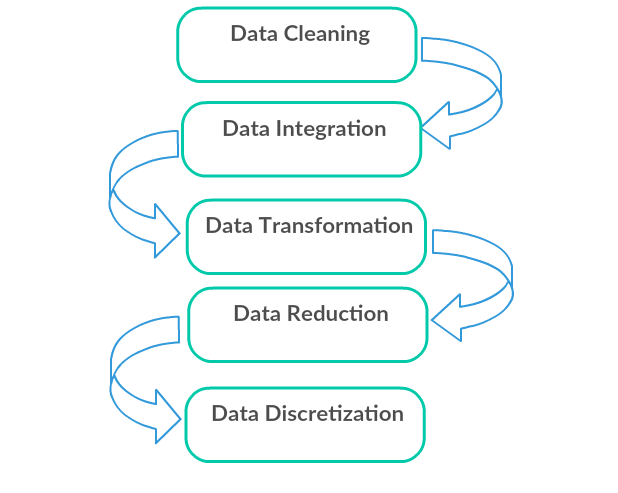
**Dataset Information looks as follows-**

****

All the data is float type so we have to convert data into int type.

* Data Pre-processing Done

Data pre-processing can refer to the manipulation or dropping of data before it is used to ensure or enhance performance, and is an important step in the data mining process.



1. Data Cleaning: First we clean the data which have no use in prediction like the S.no column and Location, then we drop the data which has a high no of missing percentages.
2. Data Integration: then we do some EDA process for finding out the meaning full insights of the data.
3. Data transformation is the process of changing the format, structure, or values of data; we use a labeled encoder for coding the object data into integer data.
4. Data Reduction: it is the process of finding the most correlated columns, and combining them because the machine does not understand which feature columns impact the most on accuracy.
5. Data discretization converts a large number of data values into smaller once, so that data evaluation and data management becomes very easy, using box plots is makes a clear understanding of the data.

* **State the set of assumptions (if any) related to the problem under consideration**

As we are trying to predict a used car price as we all know more the car is old lesser its valuation no matter what, how many features a car has.

* **Hardware and Software Requirements and Tools Used**

**Python**

Python is widely used in scientific and numeric computing:

SciPy is a collection of packages for mathematics, science, and engineering.

Pandas are data analysis and modeling libraries.

Libraries Used for this Project include –

1. Pandas

2. NumPy

3. Matplotlib

4. Seaborn

5. Scikit Learn

6. Plotly

**Model/s Development and Evaluation**

* **Identification of possible problem-solving approaches (methods)**

After analyzing the dataset, I observe that many of the feature columns are object types so first, we have to convert them into integer or float types so that the machine interprets the data and for that we do label encode all the features column.

After label encoding, we find that many feature columns have Nan values so we remove them completely because we do valuation of a used car.

Then find the correlation between the columns with target columns and delete the non-related feature columns.

We observe that the target column is skewed so we remove the skewness of the target column because normal data gives better results when we make the M.L model.

The target column is continuous type so we start work on Regression models building.

* **Testing of Identified Approaches (Algorithms)**

1. Logistic Regression
2. Regurgitation:

Lasso regression

Ridge Regression

3. Decision Tree Regression

4. Ensemble techniques

Gradient Boosting Regression

Random forest Regression.

5. Support vector machine

6. K-nearest neighbors

* **Run and Evaluate selected model**

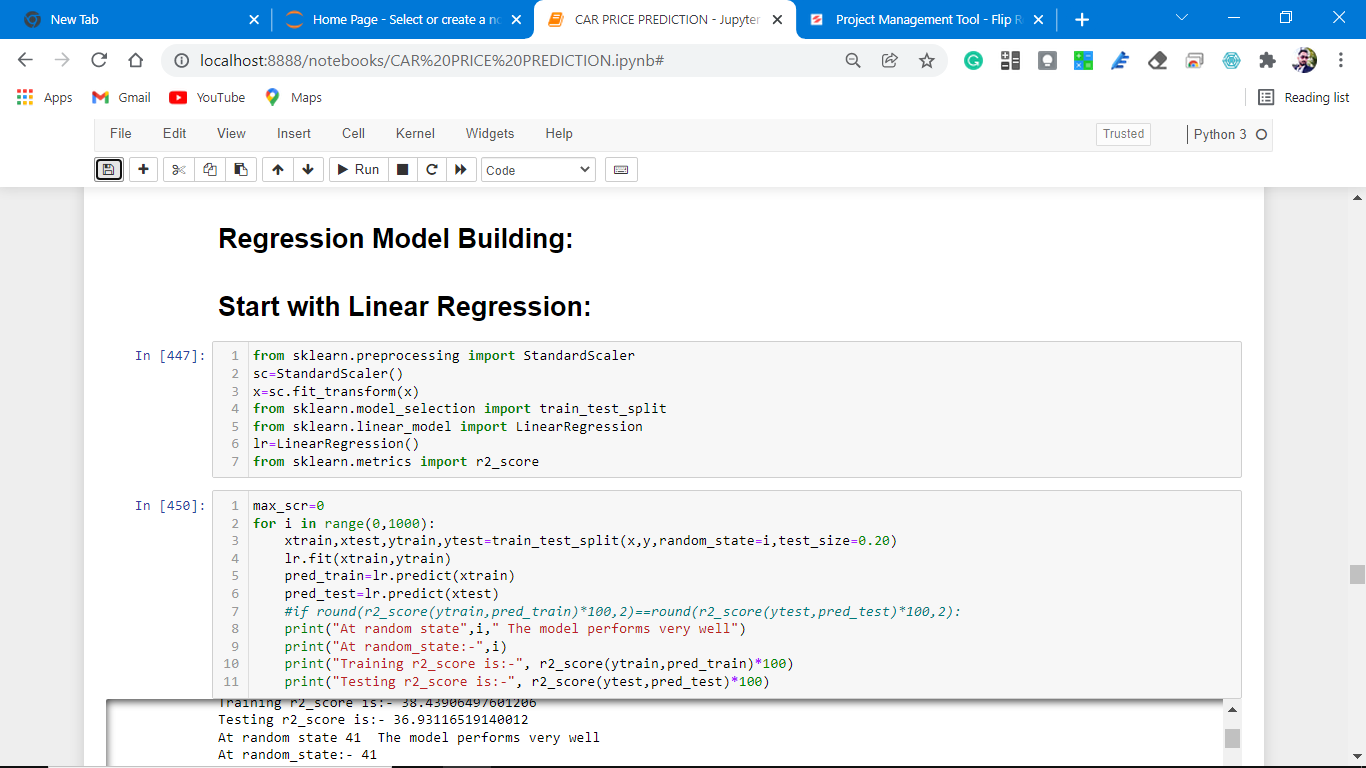
**MODELS USED**

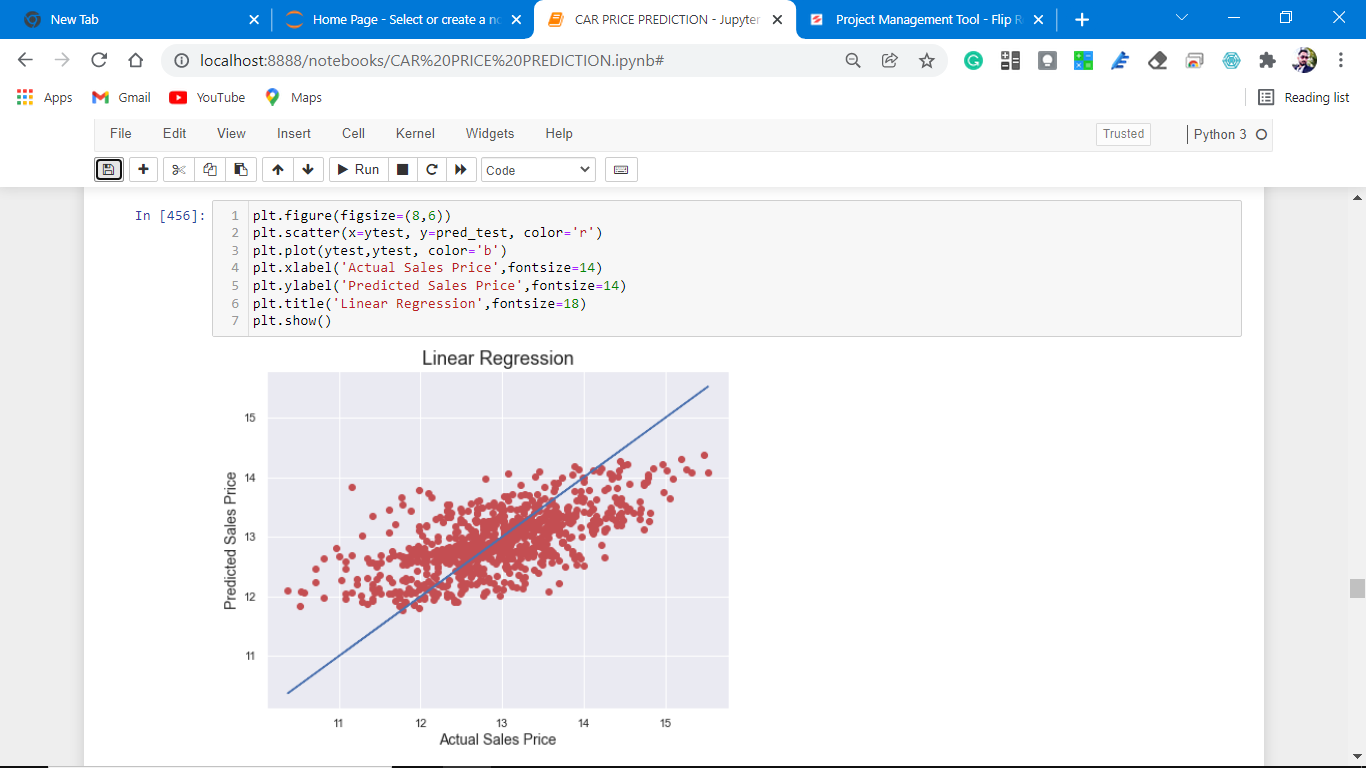
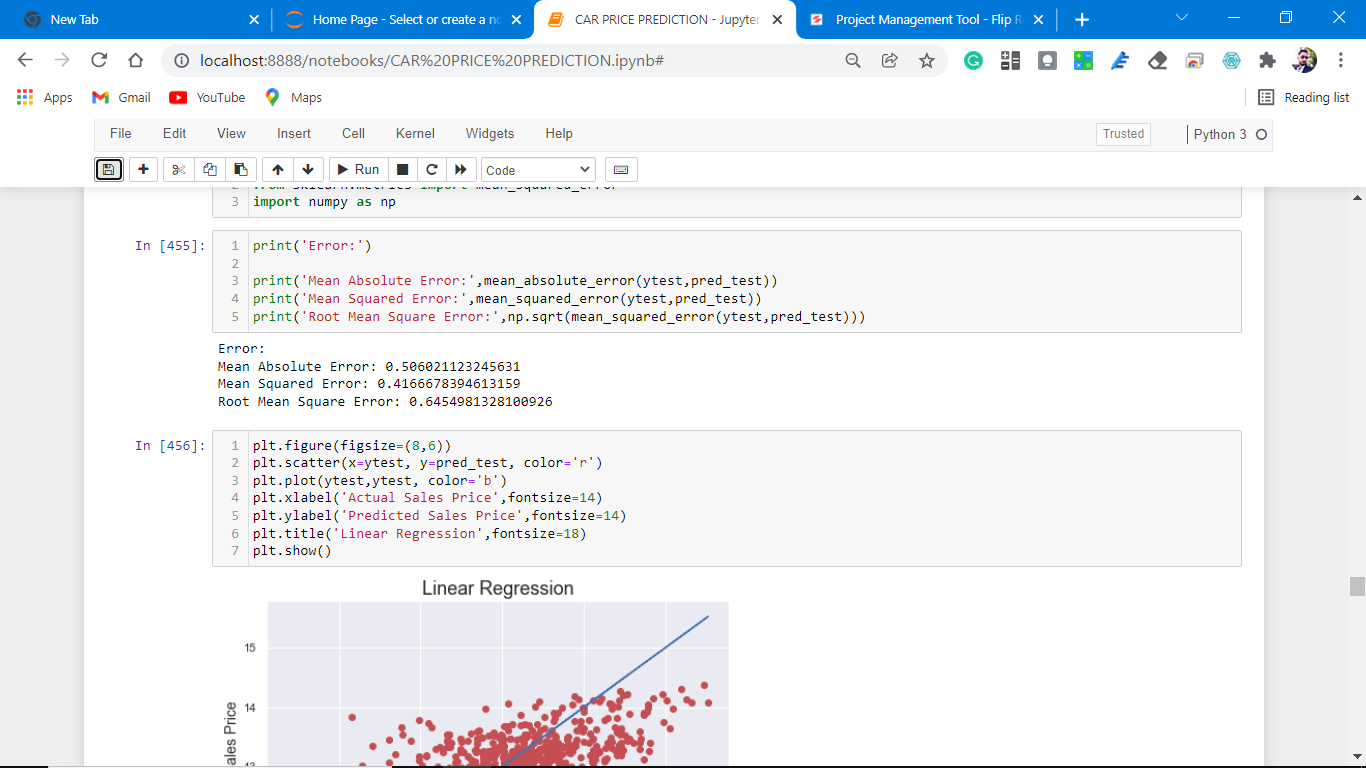
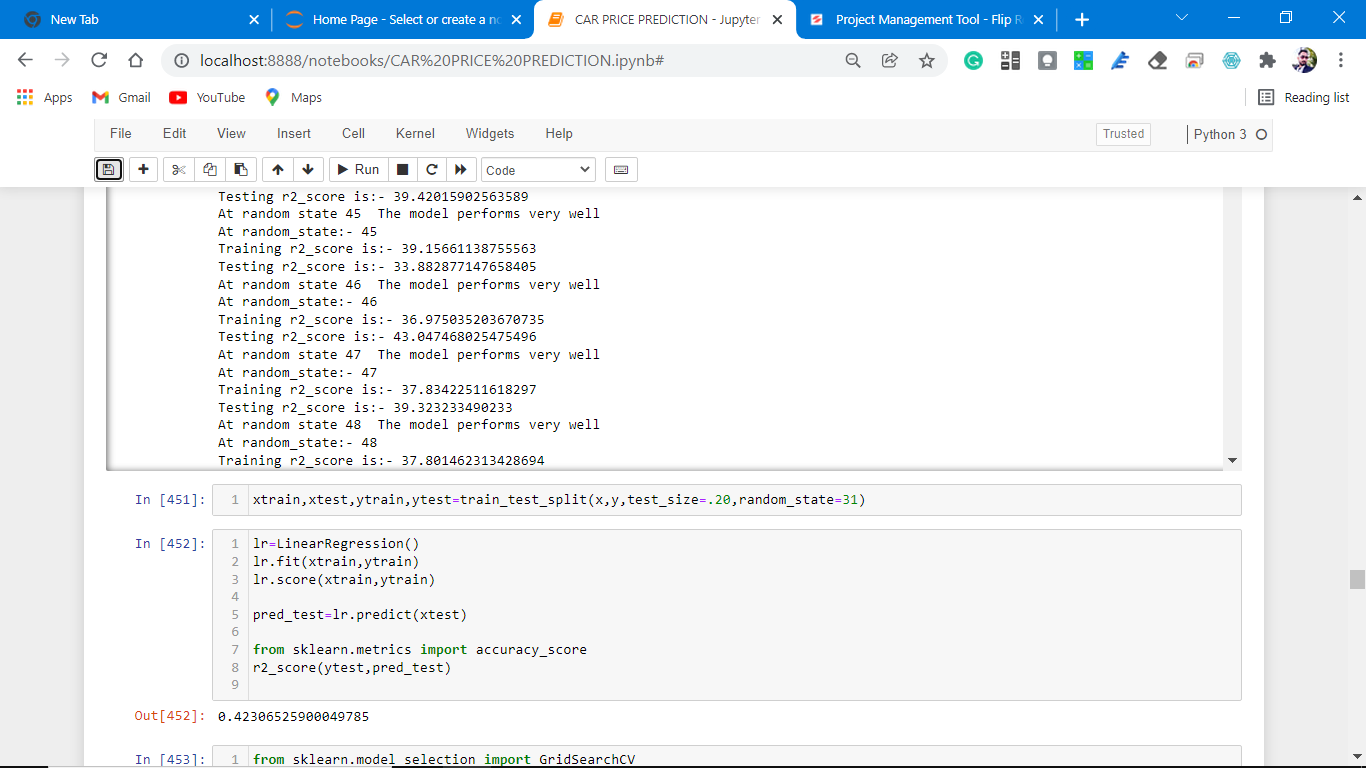
**Logistic Regression Model**

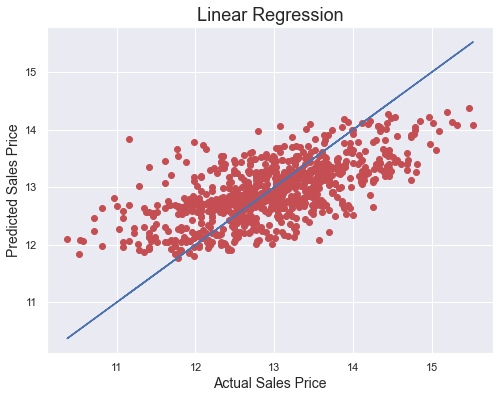
• Logistic Regression is a machine learning algorithm based on supervised learning.

• It performs a regression task. Regression models a target prediction value based on independent variables.

• It is mostly used for finding out the relationship between variables and forecasting.







**Observations:**

**1.** At random\_state:- 31

Training r2\_score is:- 37.14268657579035

Testing r2\_score is:- 42.306525900049785

The model is not so accurate; let's do some regularization.

# Regularization

# 1. Lasso Regression

# To correctly fit in our model let's do some regulation.

# The Lasso is a linear model that estimates sparse coefficients. It is useful in some contexts due to its tendency to prefer solutions with fewer non-zero coefficients, effectively reducing the number of features upon which the given solution is dependent. For this reason, Lasso and its variants are fundamental to the field of compressed sensing. Under certain conditions, it can recover the exact set of non-zero coefficients (see Compressive sensing: tomography reconstruction with L1 prior (Lasso)).

# Screenshot (1274).pngScreenshot (1275).pngScreenshot (1276).pngScreenshot (1277).pngScreenshot (1278).png

# download (1).png

# Observations:

# 1. This model is non-performing well.

# 2. There is a major difference between cross-validation and accuracy matrices.

# Cross-Validation.

At cv:- 11

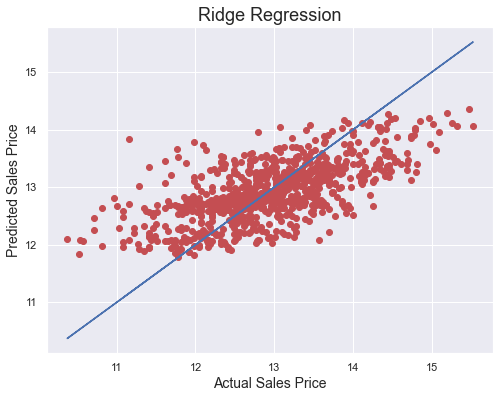
Cross-validation score is:- 33.584556993368686

R2\_score is:- 42.06893076535947

**2. Ridge Regression**

**Ridge** regression addresses some of the problems of Ordinary Least Squares by imposing a penalty on the size of the coefficients. The ridge coefficients minimize a penalized residual sum of squares:

# Screenshot (1279).pngScreenshot (1280).pngScreenshot (1281).pngScreenshot (1282).png



Observation:

1. Not performing well.

**At cv:- 10**

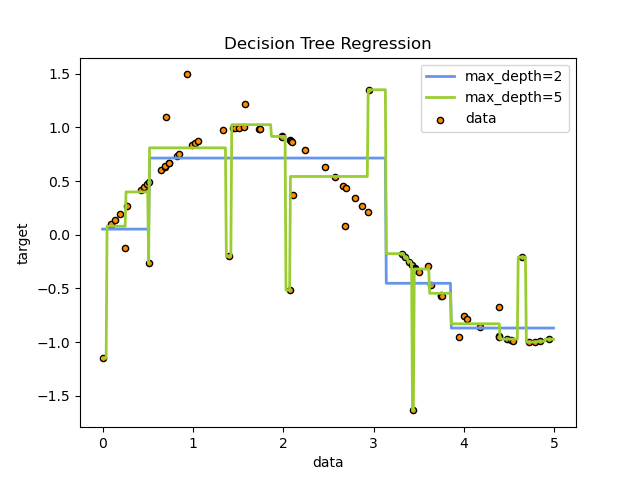
**Cross Val Score: 34.09083568790876**

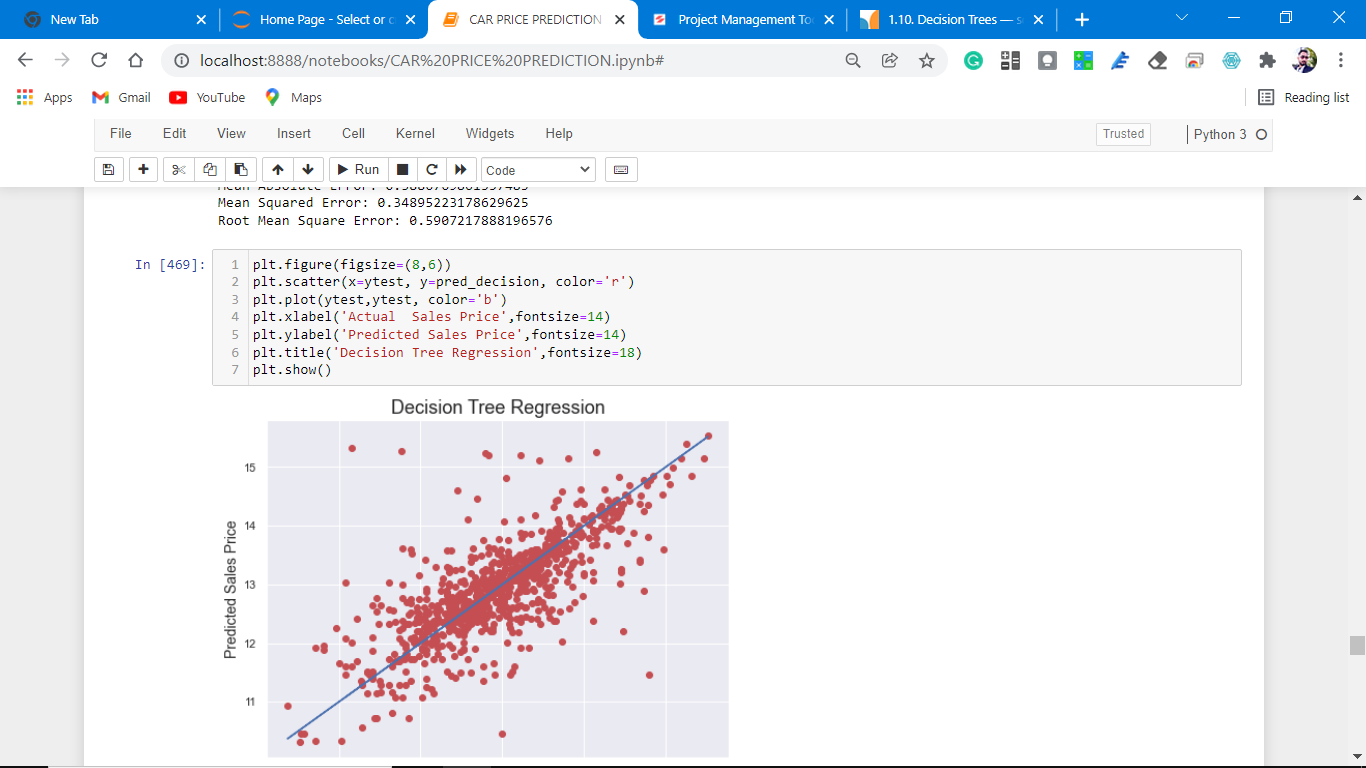
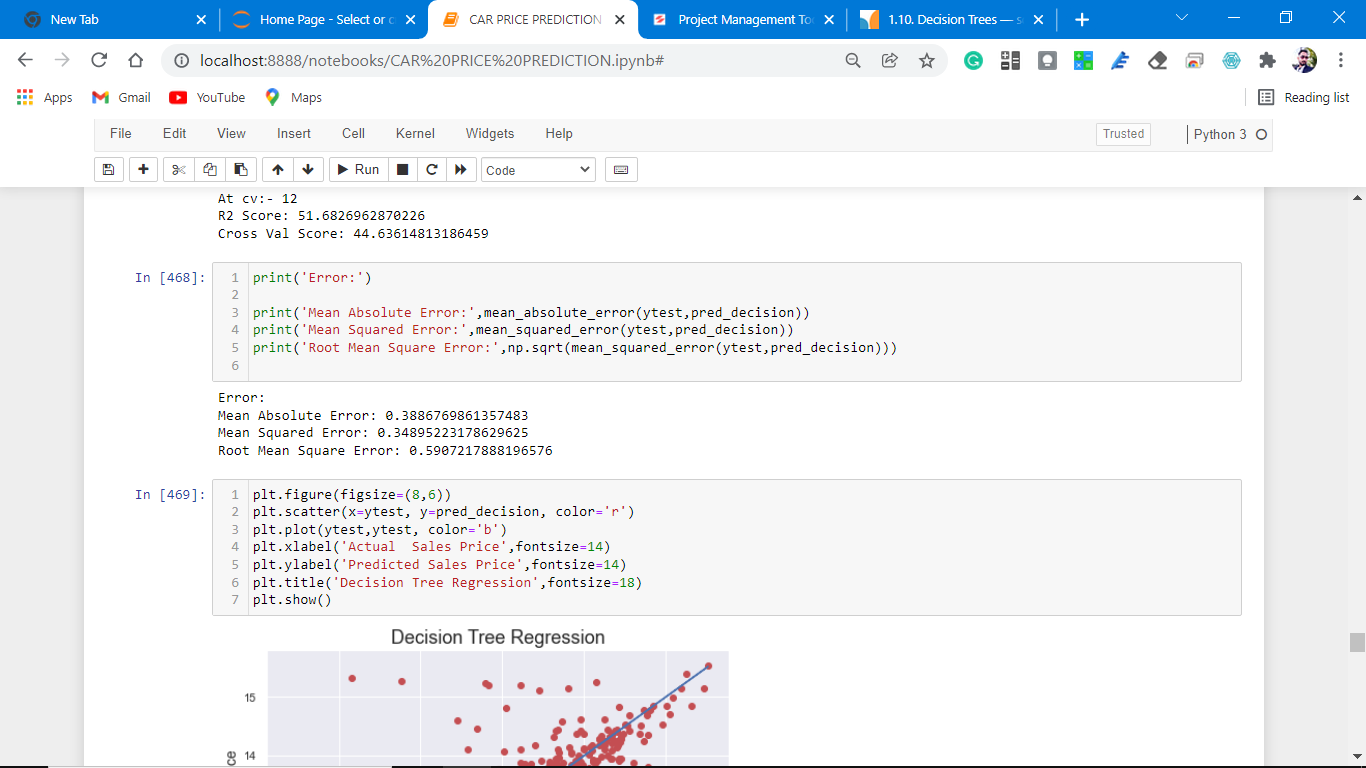
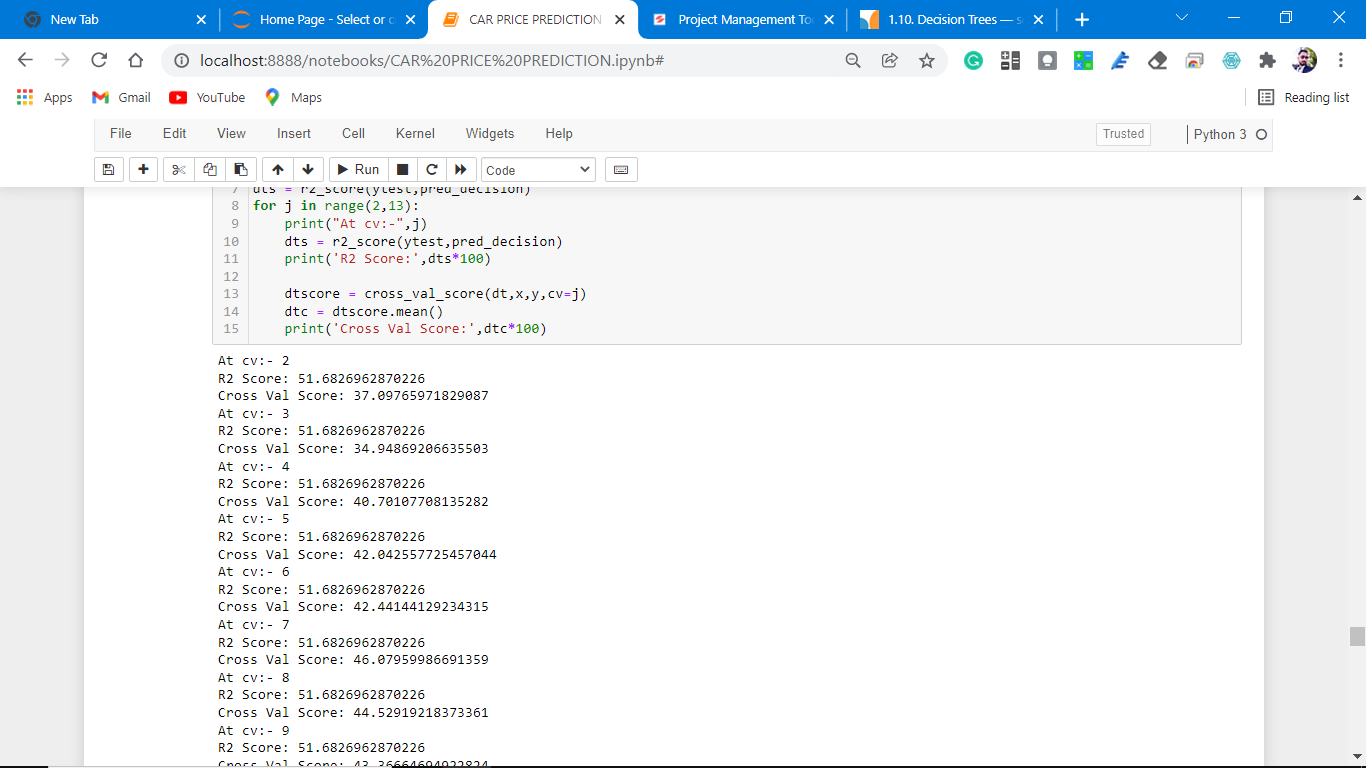
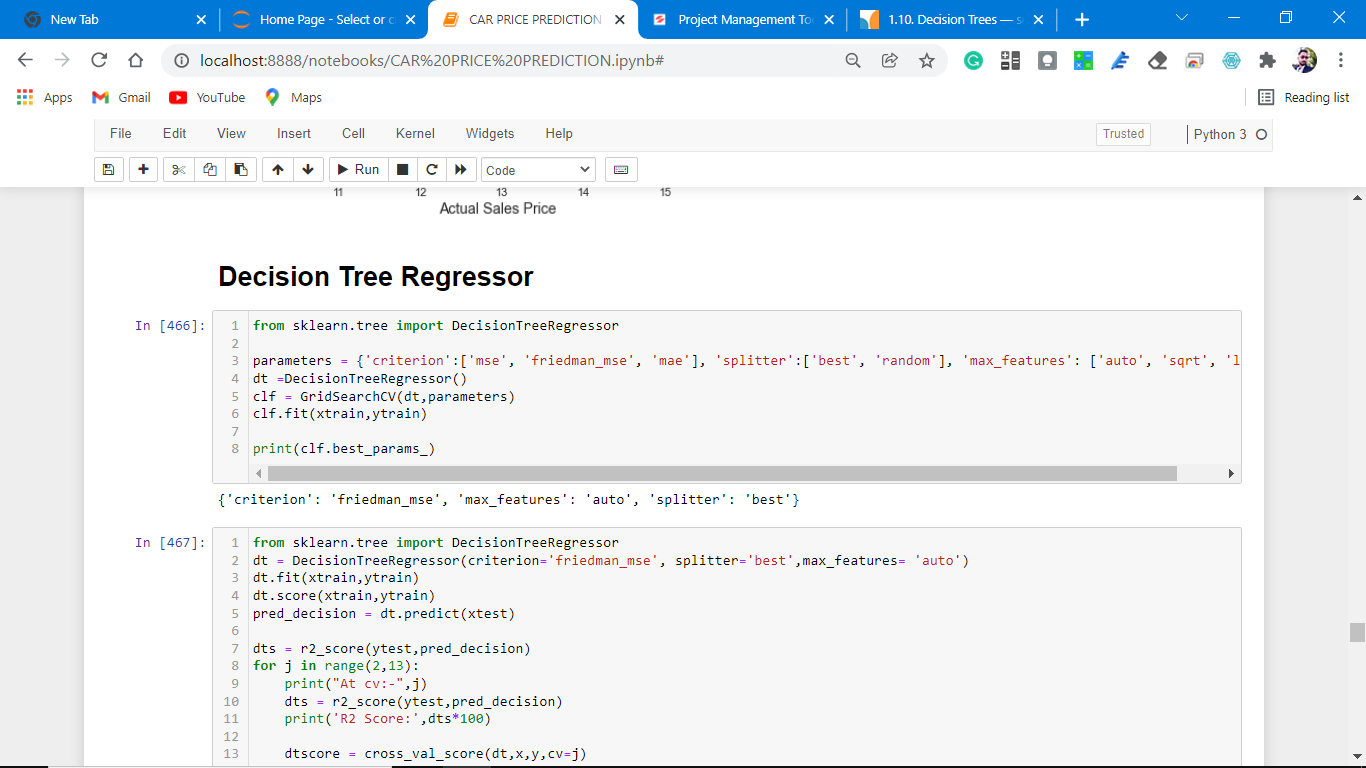
**R2 Score: 42.25130842297562**

Decision Tree Classifier

**Decision Trees (DTs)** are a non-parametric supervised learning method used for [classification](https://scikit-learn.org/stable/modules/tree.html#tree-classification) and [regression](https://scikit-learn.org/stable/modules/tree.html#tree-regression). The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

For instance, in the example below, decision trees learn from data to approximate a sine curve with a set of if-then-else decision rules. The deeper the tree, the more complex the decision rules, and the fitter the model.







**Observations:**

1. This Decision Tree regression Performs with 52% accuracy for predicting frauds.
2. After predicting and plotting the predicted data on the best fit line we observe that DT-C is not so accurate.
3. CV is not well. And does not give accurate results.

# Cross-Validation.

**At cv:- 12**

**R2 Score: 51.6826962870226**

**Cross Val Score: 44.63614813186459**

**Ensemble methods**

**1. Gradient Tree Boosting**

# Gradient Tree Boosting or Gradient Boosted Decision Trees (GBDT) is a generalization of boosting to arbitrary differentiable loss functions. GBDT is an accurate and effective off-the-shelf procedure that can be used for both regression and classification problems in a variety of areas including Web search ranking and ecology.

# Screenshot (1287).pngScreenshot (1288).pngScreenshot (1289).pngScreenshot (1290).png

# download (5).png

# Observation:

# This algorithm performs better than the above algorithms.

At cv:- 17

R2 Score: 77.94710770113188

Cross Val Score: 74.3798966811026

# 2. Random forest Tree

# The sklearn ensemble module includes two averaging algorithms based on randomized decision trees: the RandomForest algorithm and the Extra-Trees method. Both algorithms are perturb-and-combine techniques [B1998] specifically designed for trees. This means a diverse set of classifiers is created by introducing randomness in the classifier construction. The prediction of the ensemble is given as the averaged prediction of the individual classifiers.

# Screenshot (1292).pngScreenshot (1293).png

# download (6).png

# Observation:

# Perform the same as gradient boosting but not better than GBT.

# The difference between the r2 score and cv is much more than GbR.

At cv:- 4

R2 Score: 75.2257499322031

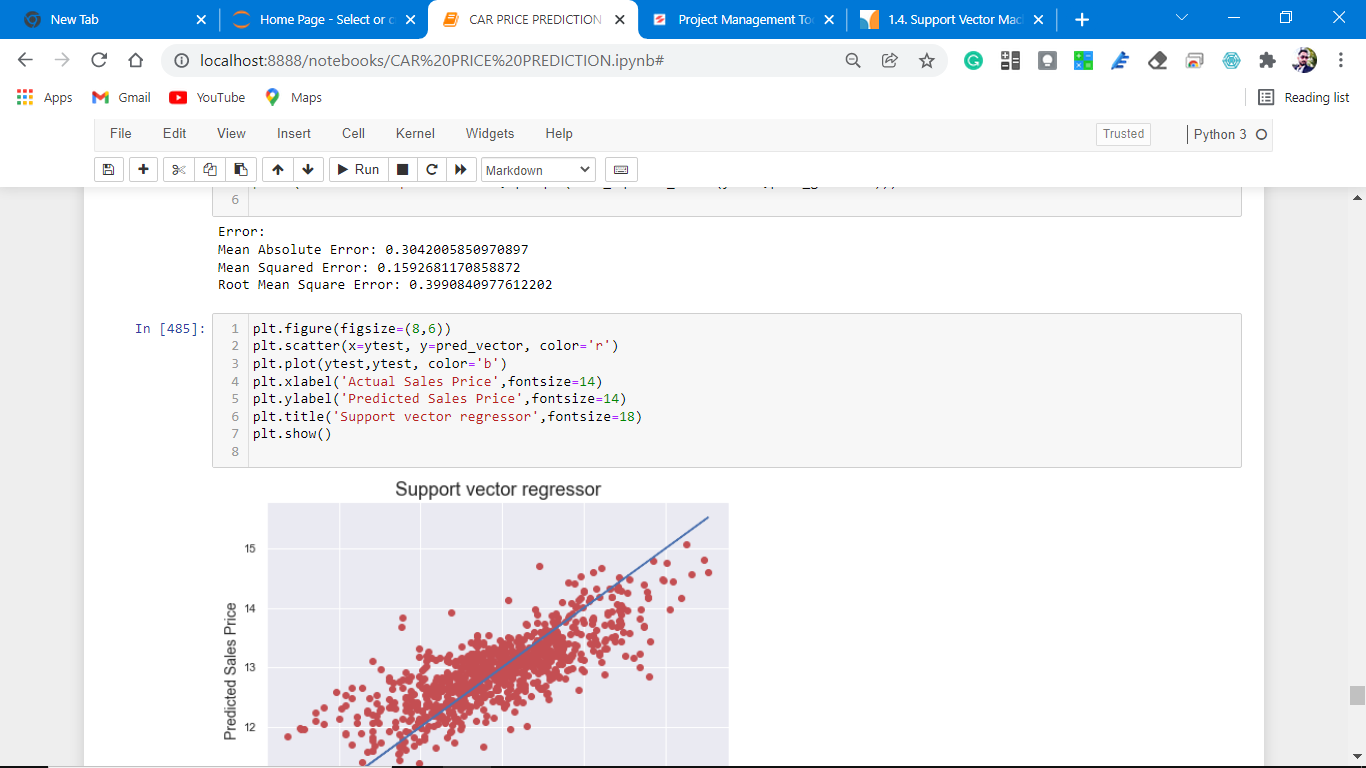
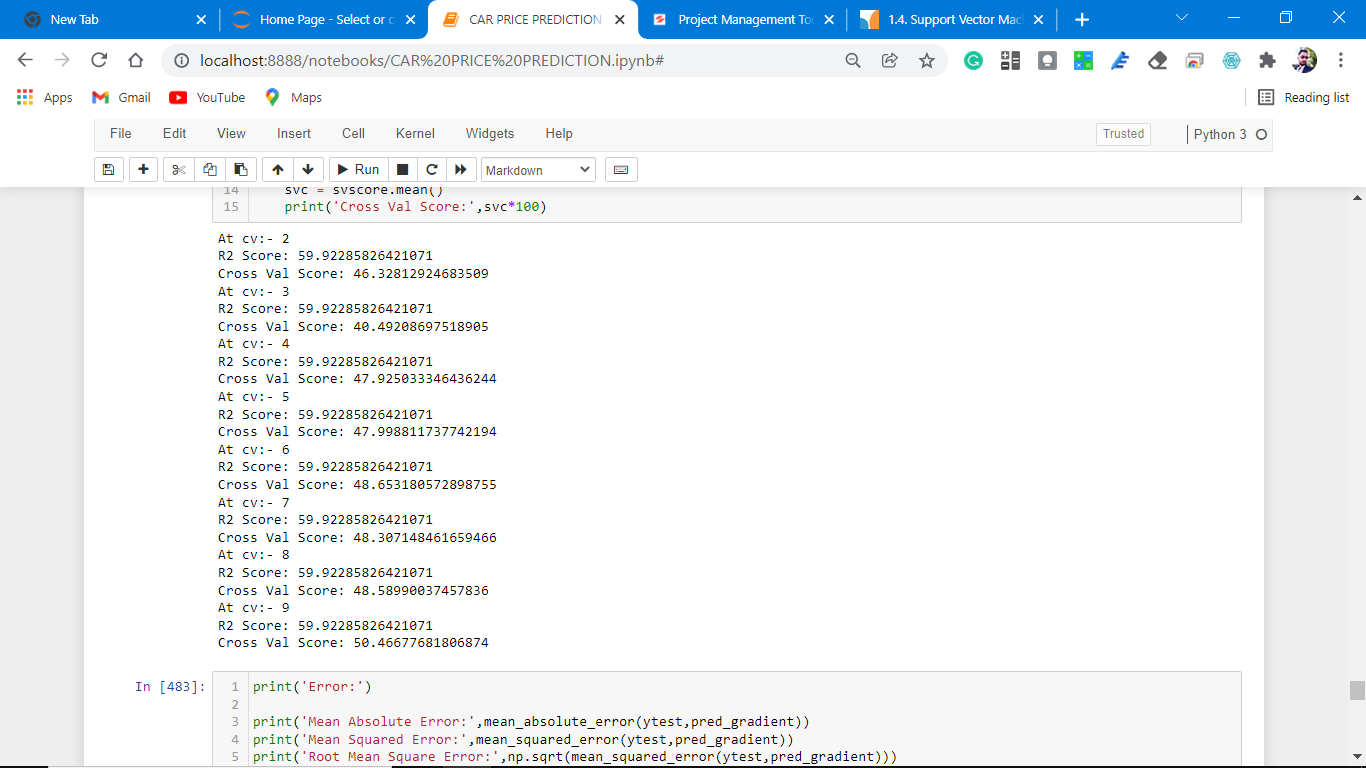
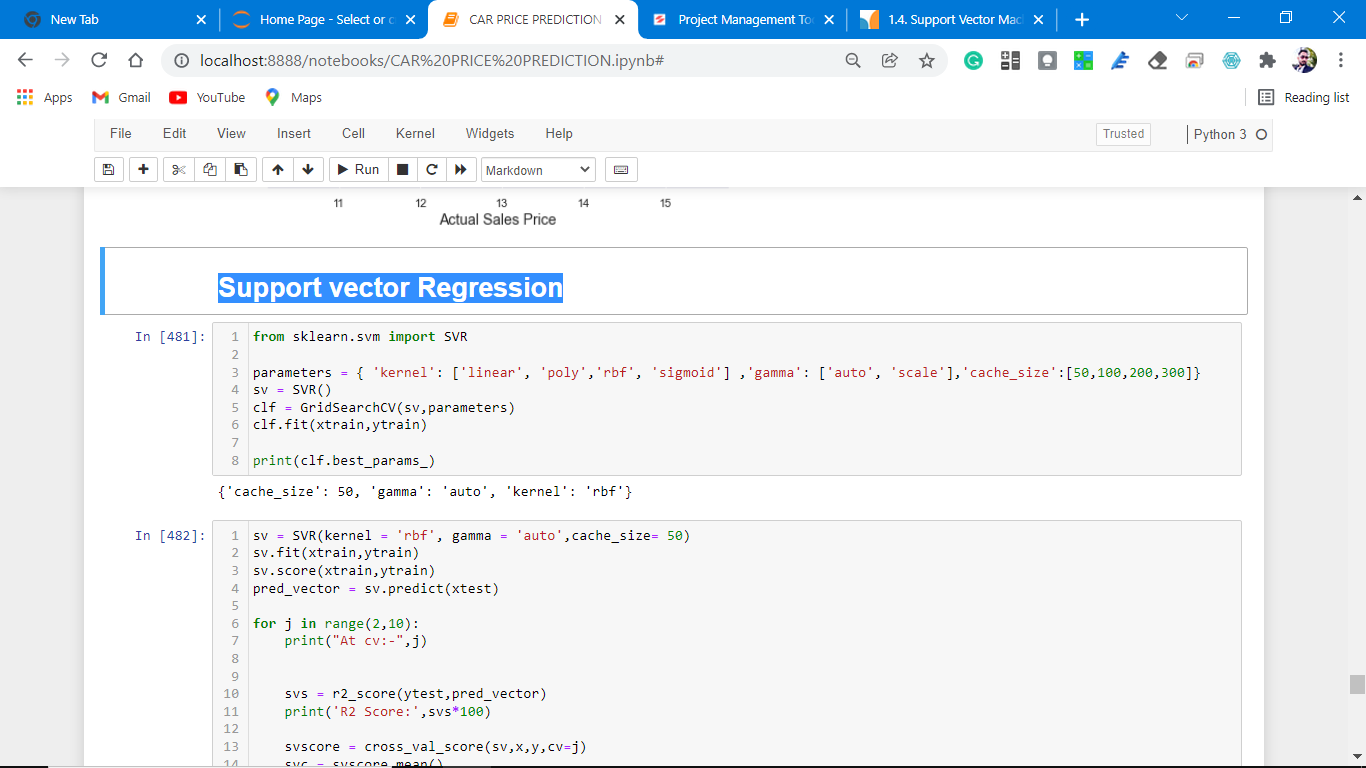
Cross Val Score: 66.56365648003803

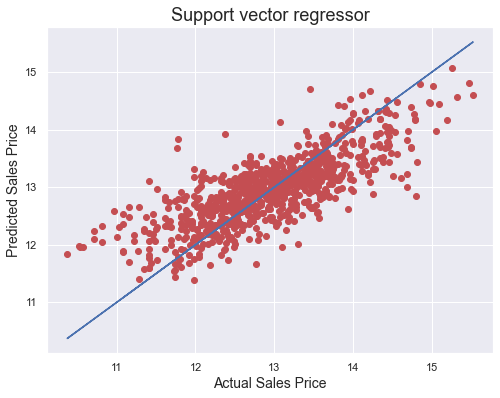
**7. Support Vector Machine.**

The method of Support Vector Classification can be extended to solve regression problems. This method is called Support Vector Regression.

The model produced by support vector classification (as described above) depends only on a subset of the training data because the cost function for building the model does not care about training points that lie beyond the margin. Analogously, the model produced by Support Vector Regression depends only on a subset of the training data, because the cost function ignores samples whose prediction is close to their target.

There are three different implementations of Support Vector Regression: SVR, NuSVR, and [LinearSVR](https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVR.html#sklearn.svm.LinearSVR). [LinearSVR](https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVR.html#sklearn.svm.LinearSVR) provides a faster implementation than SVR but only considers the linear kernel, while NuSVR implements a slightly different formulation than [SVR](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html#sklearn.svm.SVR) and [LinearSVR](https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVR.html#sklearn.svm.LinearSVR). See [Implementation details](https://scikit-learn.org/stable/modules/svm.html#svm-implementation-details) for further details.





# Observation:

# 1. this algorithm does not perform as expected.

At cv:- 9

R2 Score: 59.92285826421071

Cross Val Score: 50.46677681806874

**8. KNeighborsRegressor**

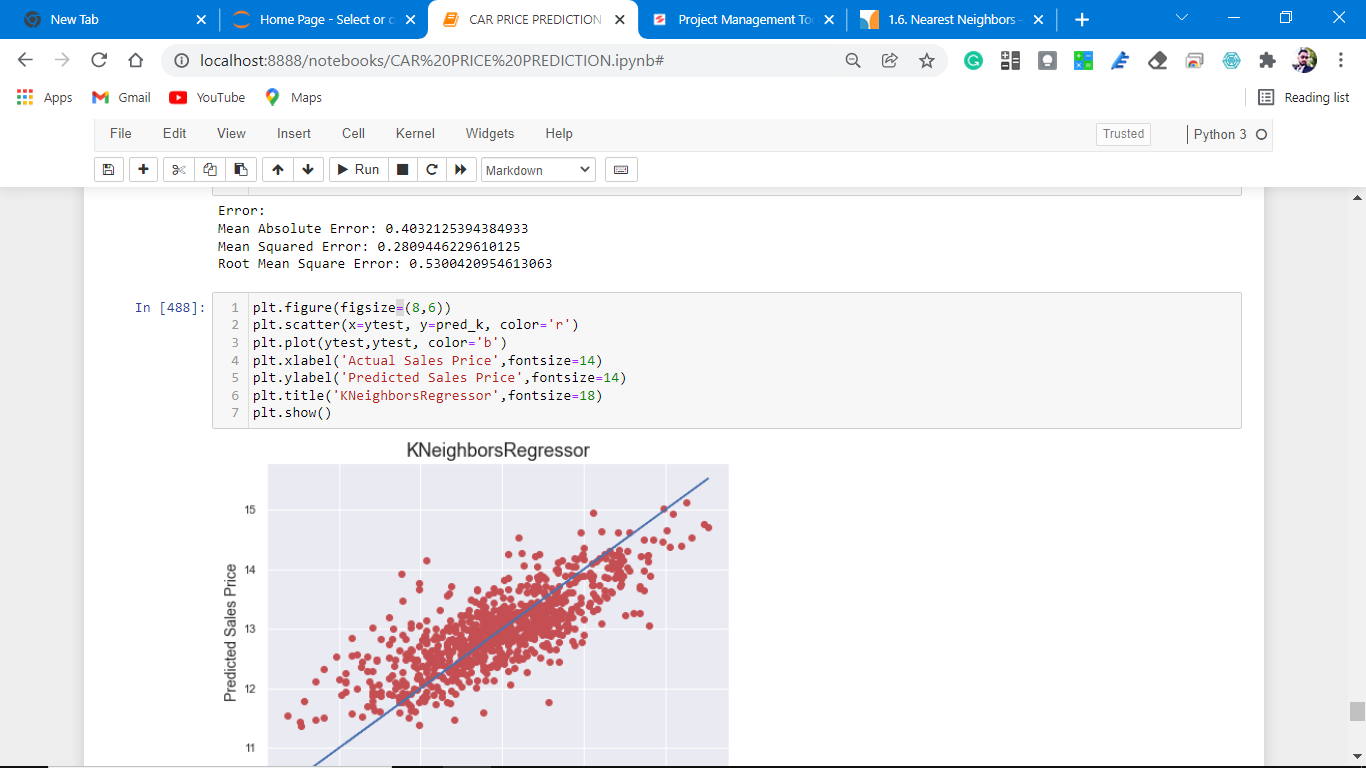
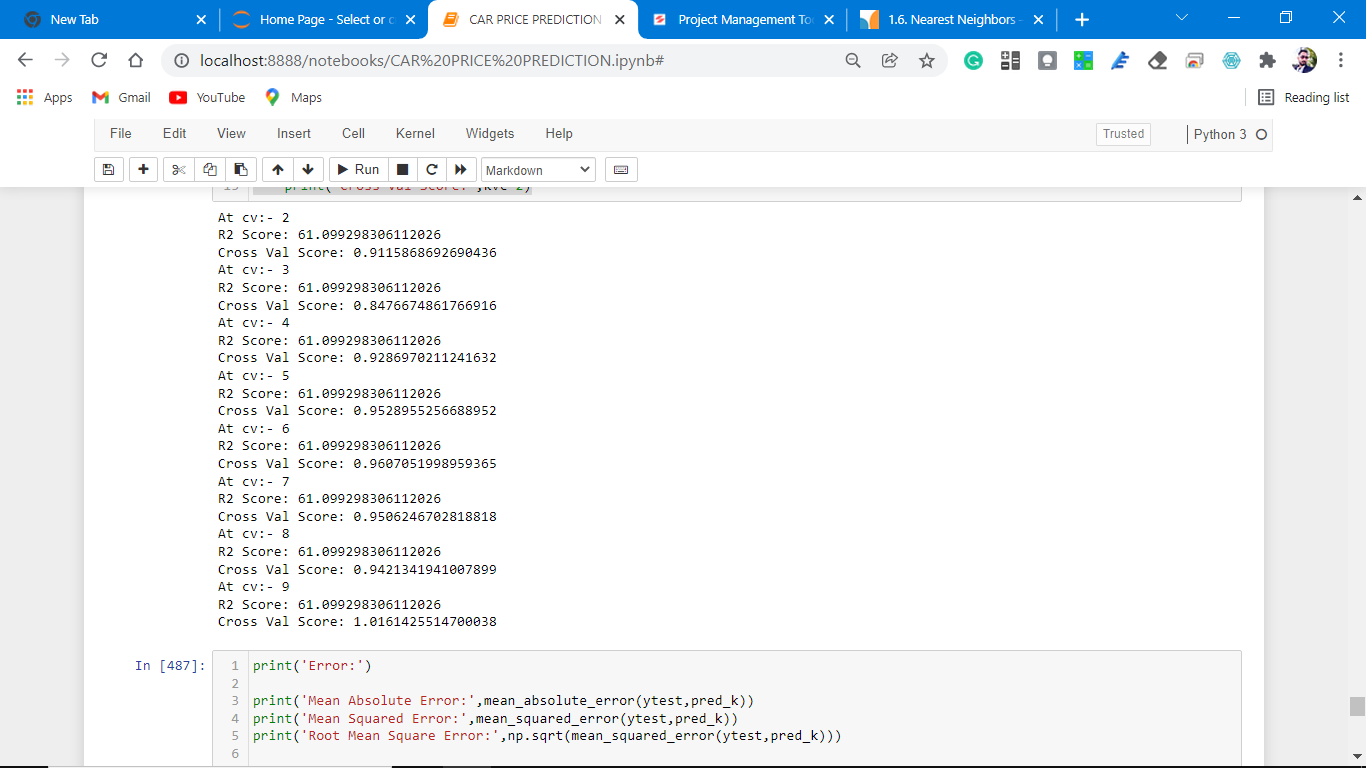
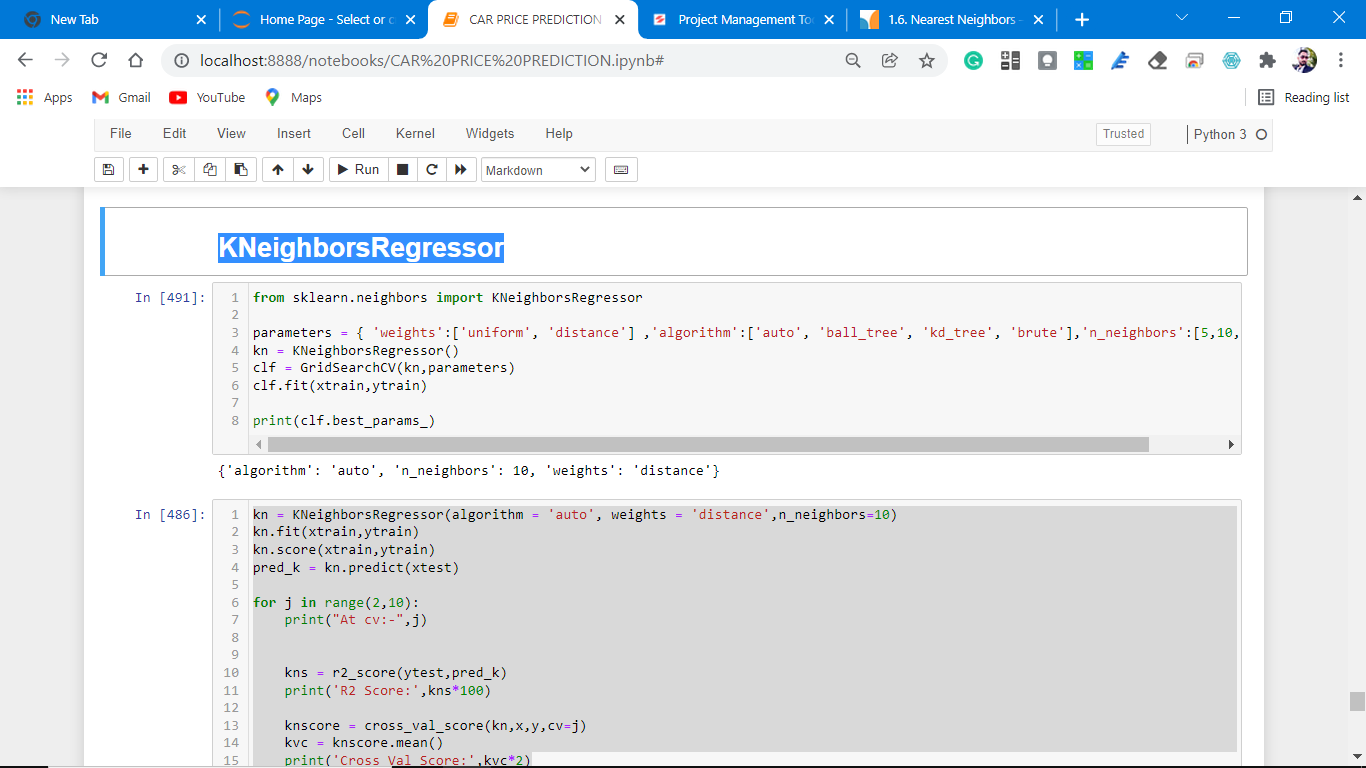
[**sklearn. neighbors**](https://scikit-learn.org/stable/modules/classes.html#module-sklearn.neighbors) provide functionality for unsupervised and supervised neighbors-based learning methods. Unsupervised nearest neighbors are the foundation of many other learning methods, notably manifold learning and spectral clustering. Supervised neighbors-based learning comes in two flavors: [classification](https://scikit-learn.org/stable/modules/neighbors.html#classification) for data with discrete labels and [regression](https://scikit-learn.org/stable/modules/neighbors.html#regression) for data with continuous labels.

The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning) or vary based on the local density of points (radius-based neighbor learning). The distance can, in general, be any metric measure: standard Euclidean distance is the most common choice. Neighbors-based methods are known as non-generalizing machine learning methods since they simply “remember” all of their training data (possibly transformed into a fast indexing structure such as a [Ball Tree](https://scikit-learn.org/stable/modules/neighbors.html#ball-tree) or [KD Tree](https://scikit-learn.org/stable/modules/neighbors.html#kd-tree)).

Despite its simplicity, nearest neighbors have been successful in a large number of classification and regression problems, including handwritten digits and satellite image scenes. Being a non-parametric method, it is often successful in classification situations where the decision boundary is very irregular.

The classes in sk[**learn. neighbors**](https://scikit-learn.org/stable/modules/classes.html#module-sklearn.neighbors) can handle either NumPy arrays or scipy.sparse matrices as input. For dense matrices, a large number of possible distance metrics are supported. For sparse matrices, arbitrary Minkowski metrics are supported for searches.

There are many learning routines that rely on nearest neighbors at their core. One example is [kernel density estimation](https://scikit-learn.org/stable/modules/density.html#kernel-density), discussed in the [density estimation](https://scikit-learn.org/stable/modules/density.html#density-estimation) section.



# download (8).png

# Observation:

# 1. we see that not perform at all.

At cv:- 9

R2 Score: 61.099298306112026

Cross Val Score: 1.0161425514700038

* Key Metrics for success in solving a problem under consideration

Mean Absolute Error, Mean Squared Error, Root Mean Square Error

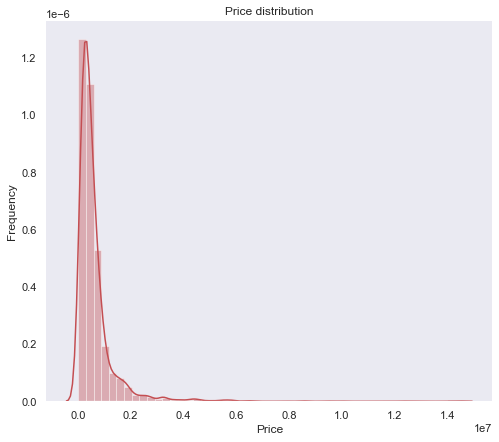
This matrix helps to understand the model more deeply.

* **Visualizations**

Data visualization is the graphical representation of information and data. By using charts, plots, and graphs data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.

**Starts from label column analysis.(target variable)**

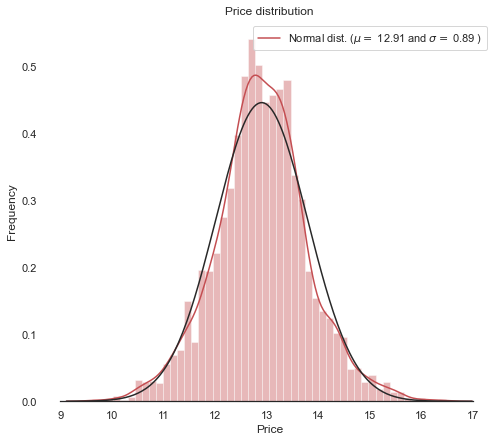


**Observations:**

1. As seen from the above distribution plot that data is not normally distributed.

2. data is skewed towards the right.

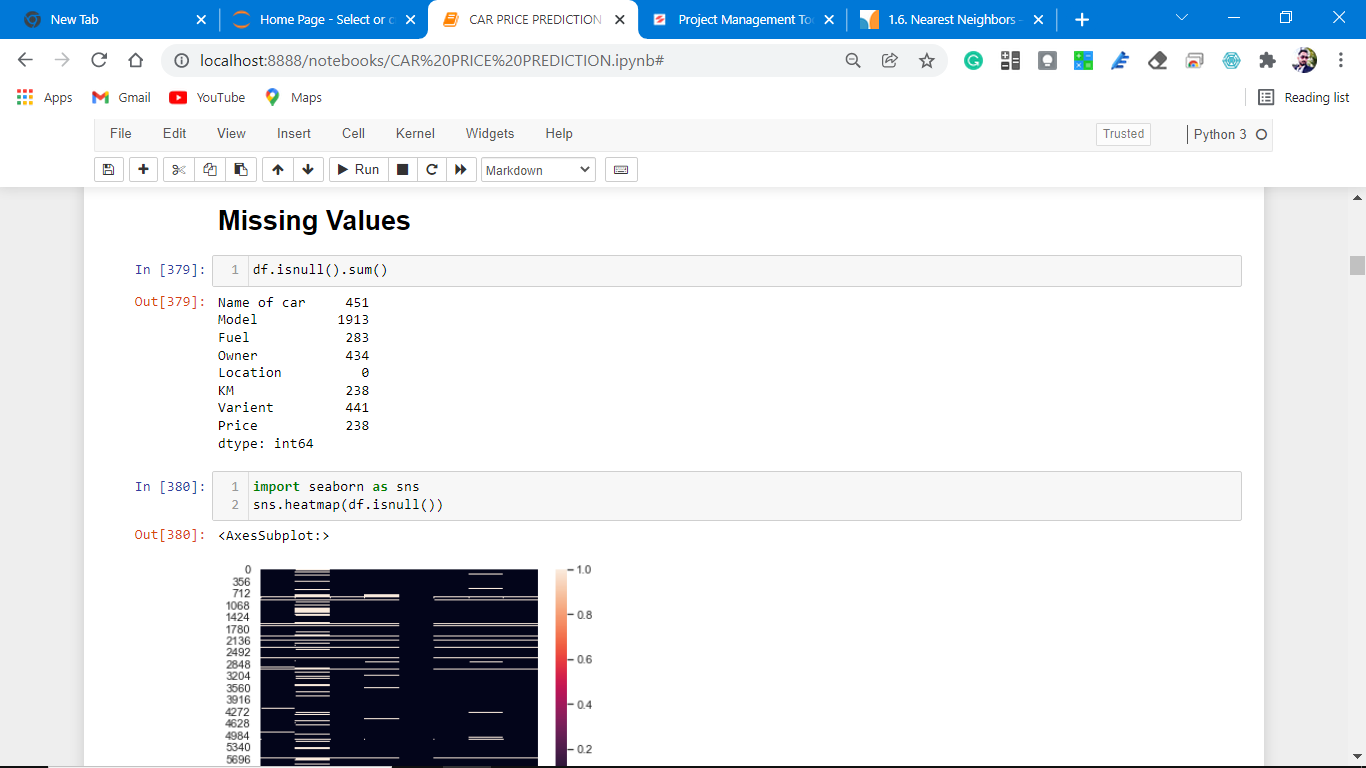
Remove the skewness to make data normally distributed, M.L models understand normally data more clearly and perform better prediction.

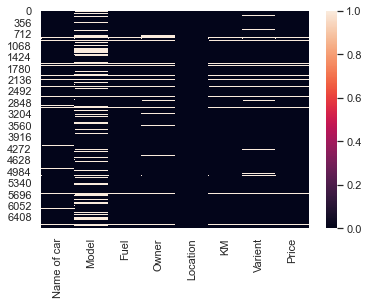


Observations:

1. As we see our price attribute is now normally distributed.

**Heat Map Plotting for null values.**

****

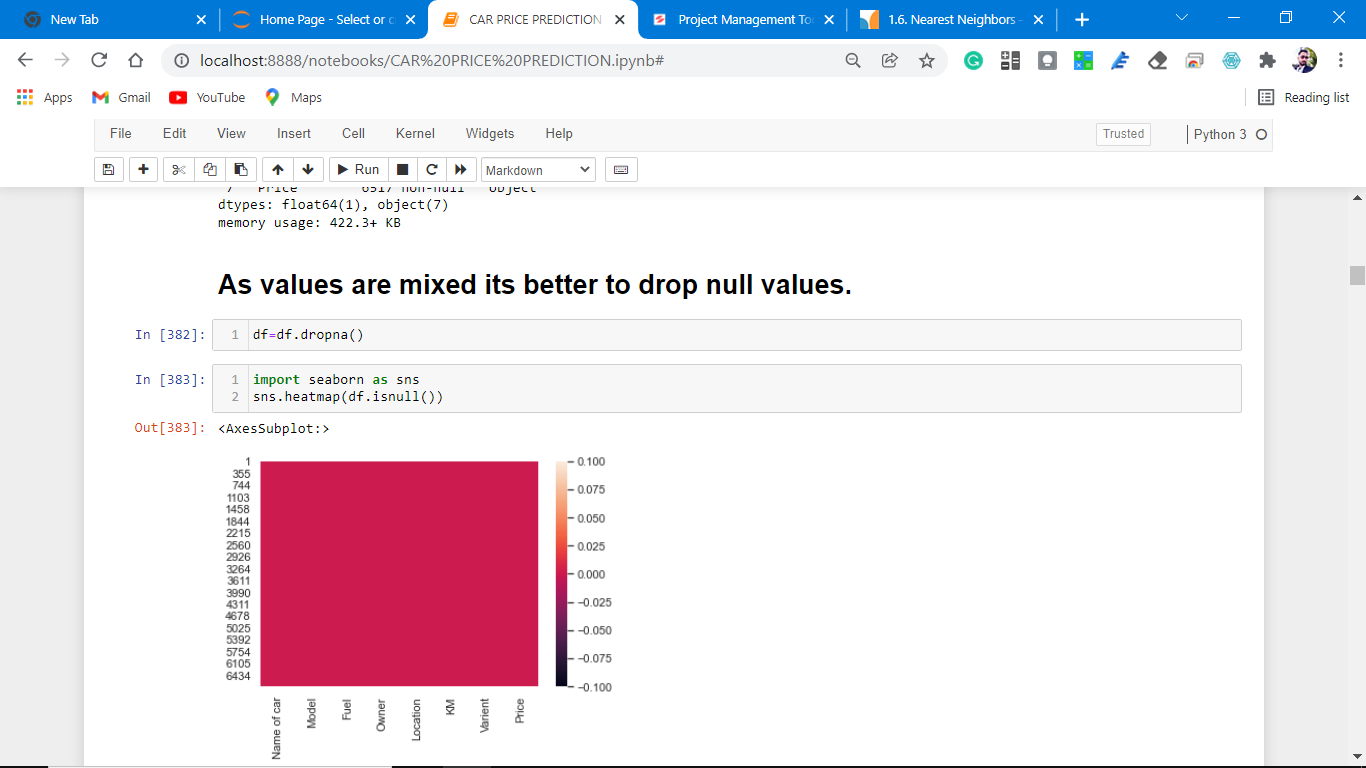
****

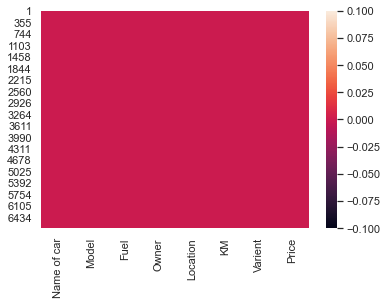
**Observations:**

1. From the above code we observe that there are lots of white spots which are null values in the dataset.

2. we have to understand the missing values and fill with proper values or drop them permanently.

**Dropping all the null values.**

****

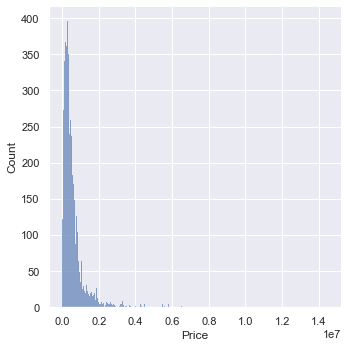
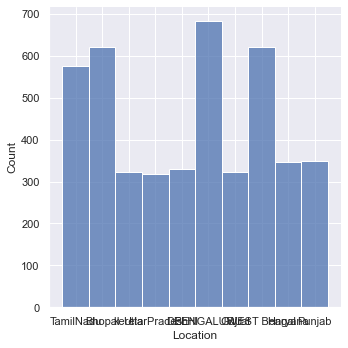
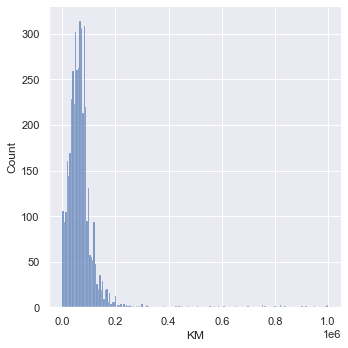
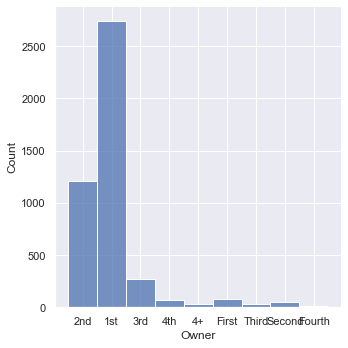
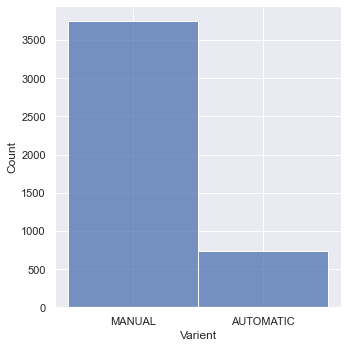
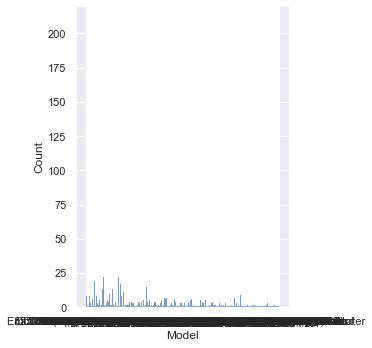
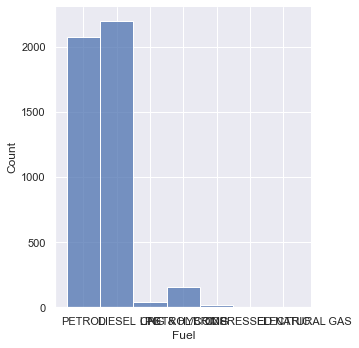


Observations:

1. As observed from the above there is no null values count

, we can easily visualize this heat map plotting.

**Univariate Analysis.**



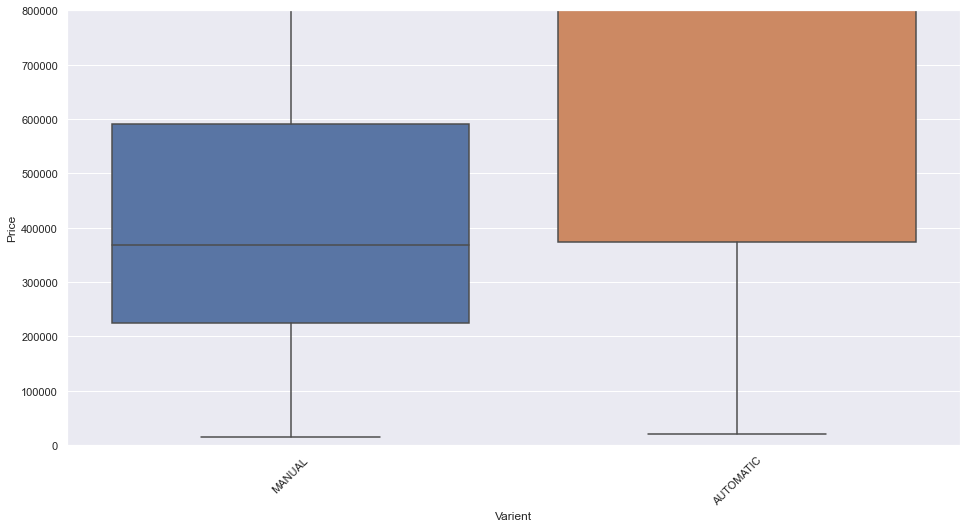
**Observations:**

1. From the above plotting of distribution plot we see that some features columns are not normally distributed.

2. some columns are skewed towards the right.

**Bivariate analysis for better understanding which features columns is impacting more on Predicting Prices.**

Price vs variant of the car.

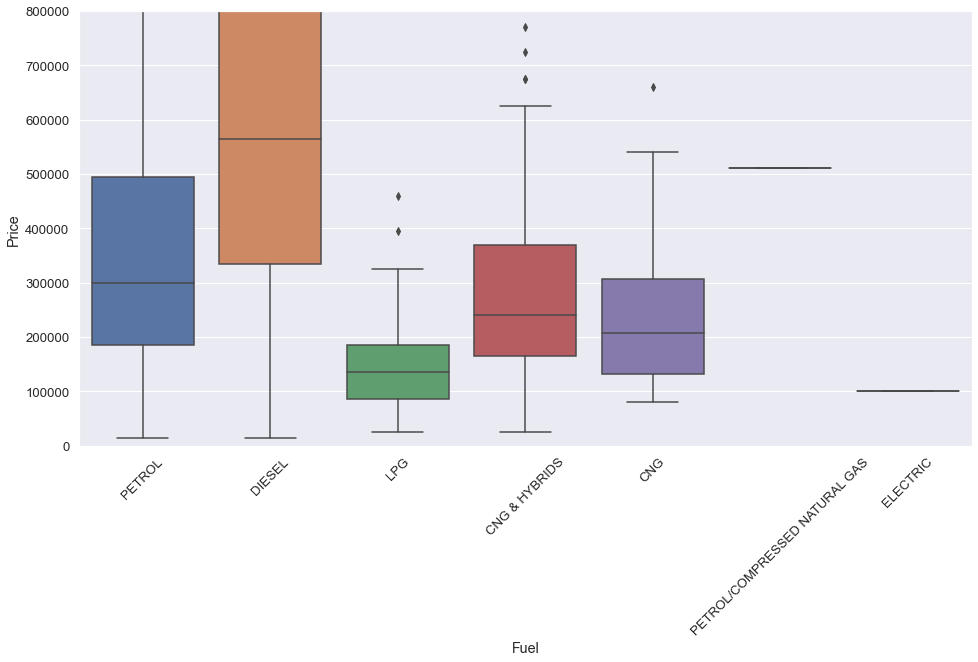
****

Observations:

1. Classes are not equal.

2. Both the segments have different price bands, automatic vehicles are more expensive than the manual transmission.

Price vs fuel options.

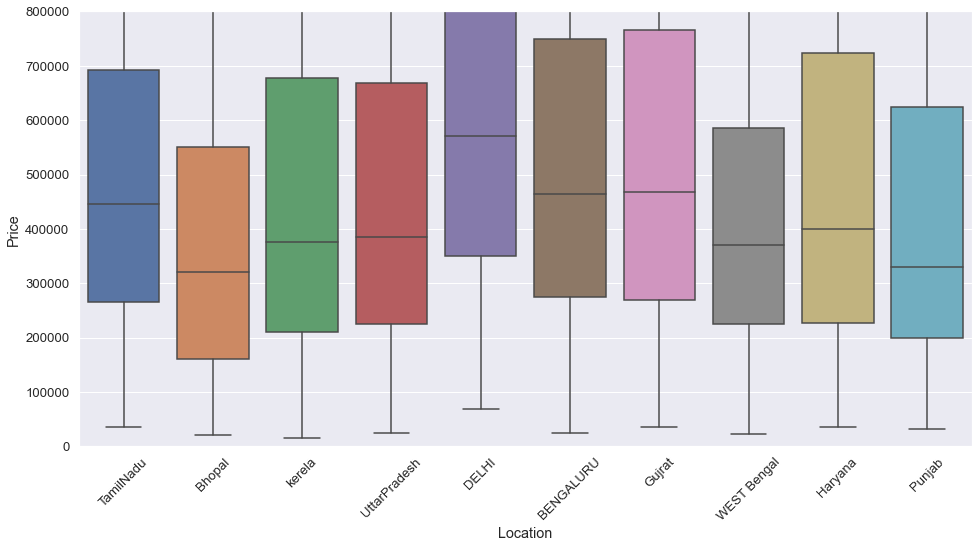


Observations:

1. Diesel car having more price range.

2. petrol vehicles on 2nd spot in the price range.

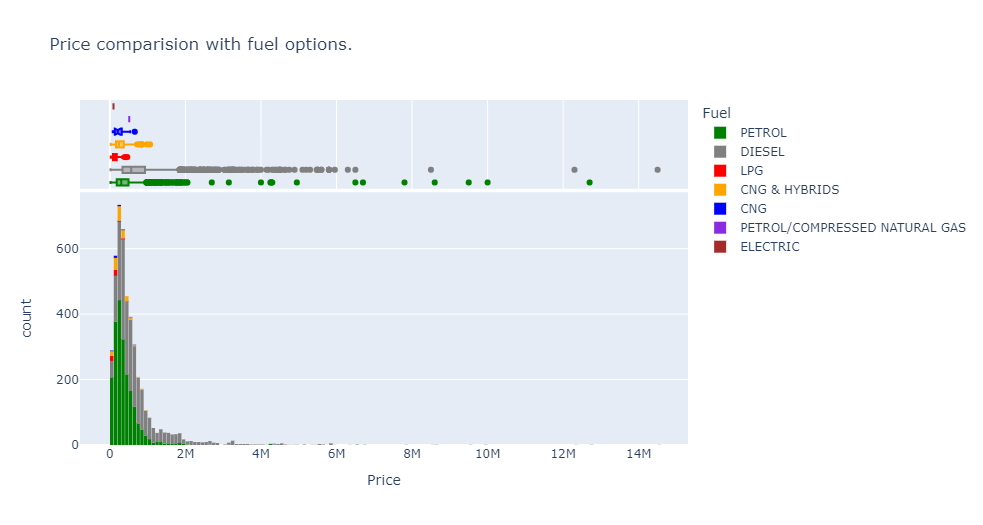
3. CNG and hybrid cars are in the 3rd spot in the price range.

****

**Price with Owner**

****

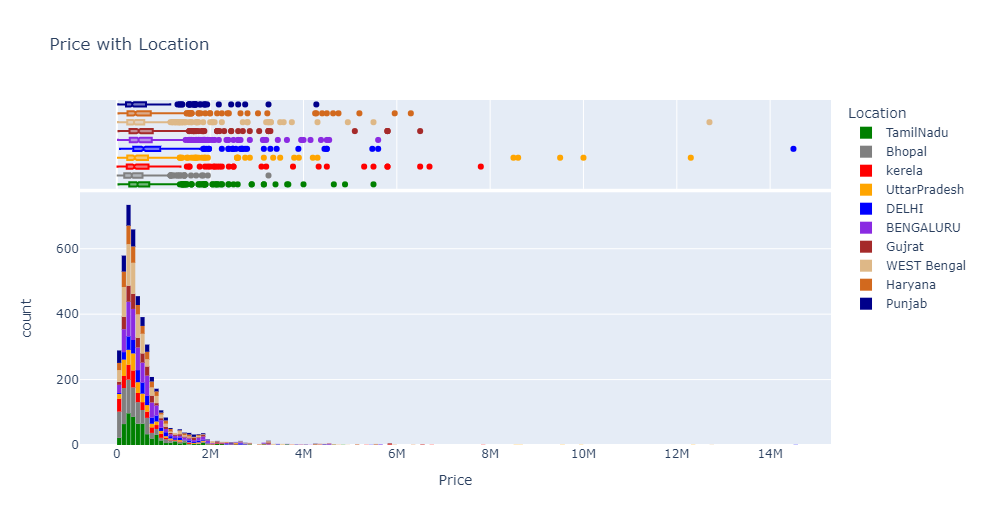
**Price comparison with fuel options**

****

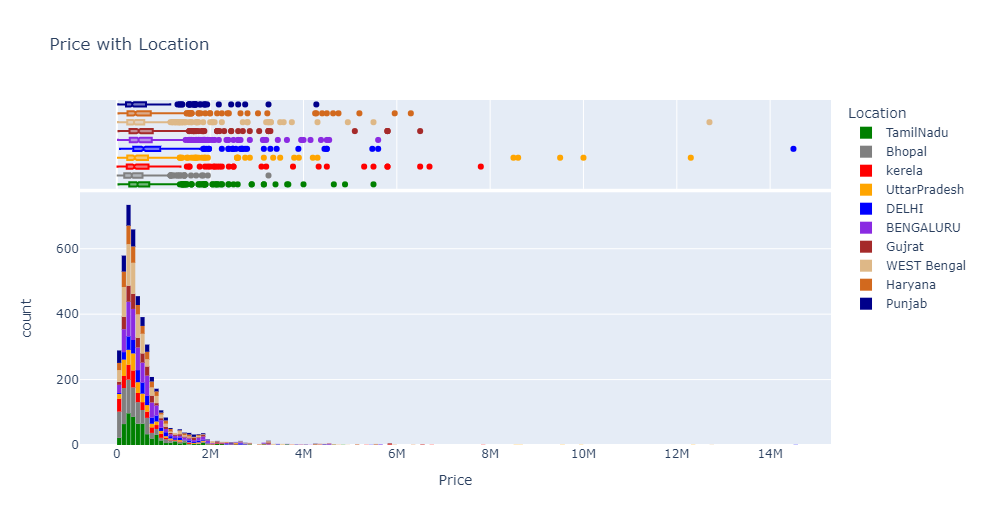
**Price with variant**

****

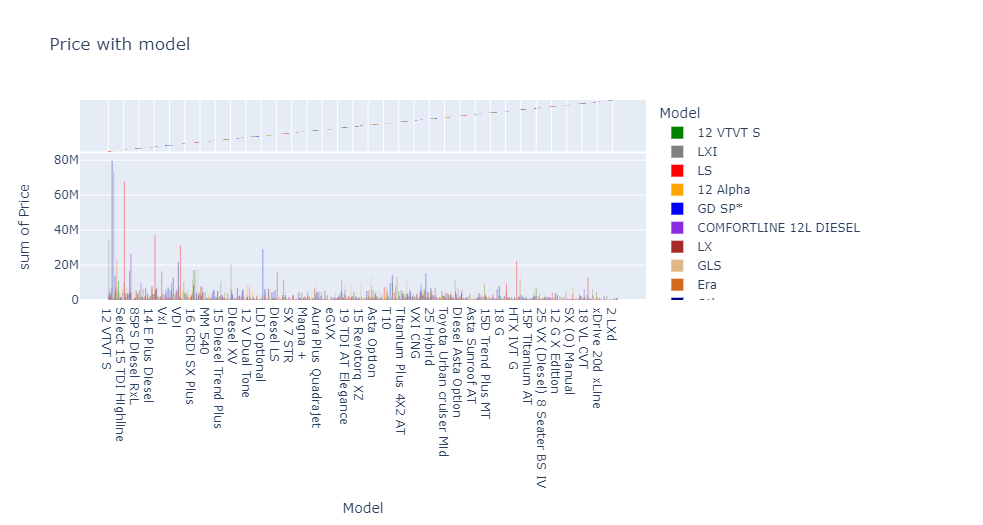
**Price with Location**

****

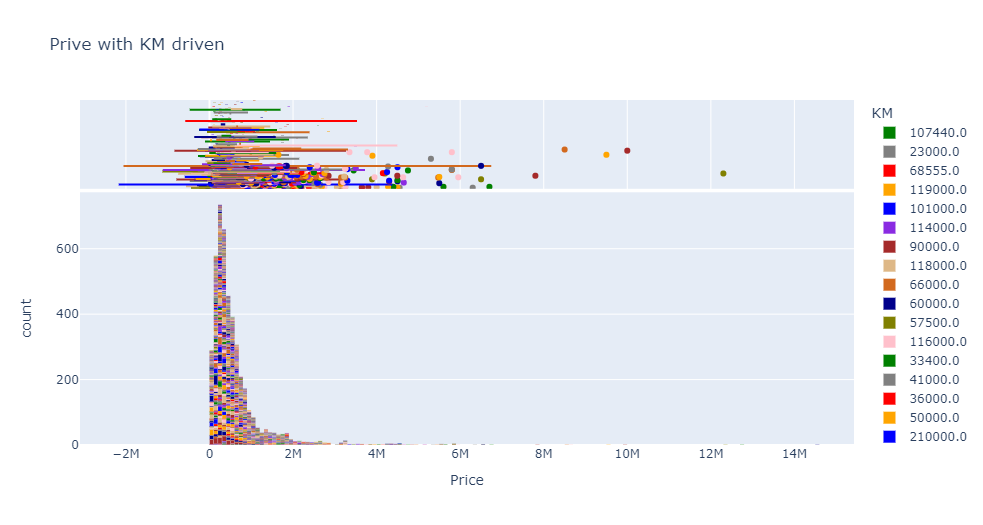
**Cars with variant**

****

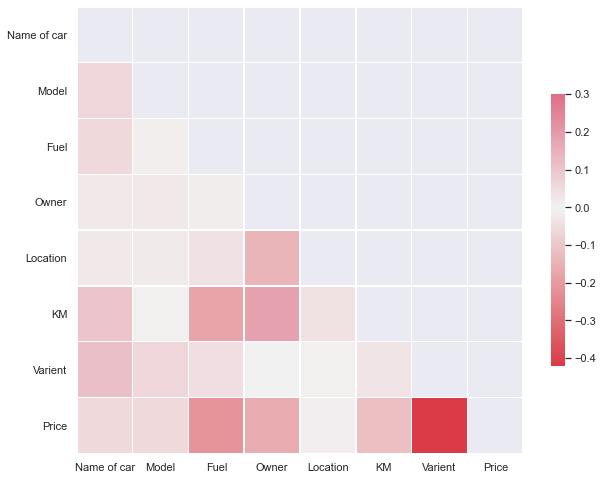
**Price with model**

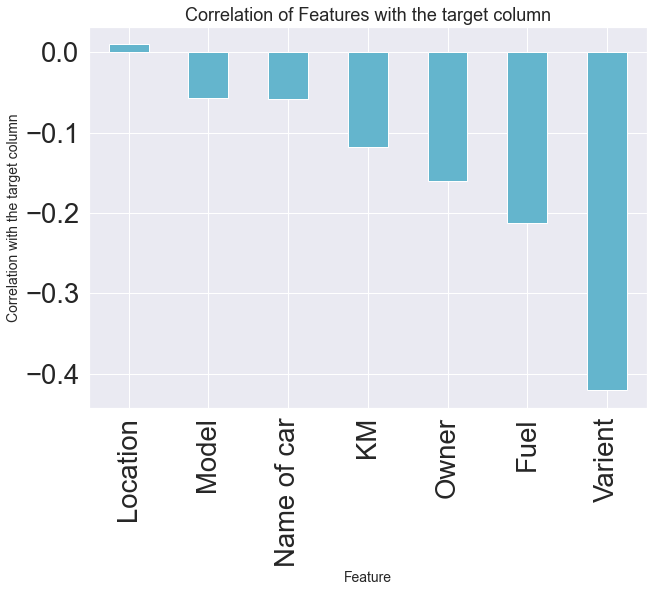
****

**Price with km driven.**

****

**Correlation with the target variable.**

****

****

Observations:

1. From the above result it is clear that some columns make a positive correlation and some make a negative correlation.

2. The positively correlated columns have a great impact on the target column while the negatively correlated have less or zero impact on the target column.

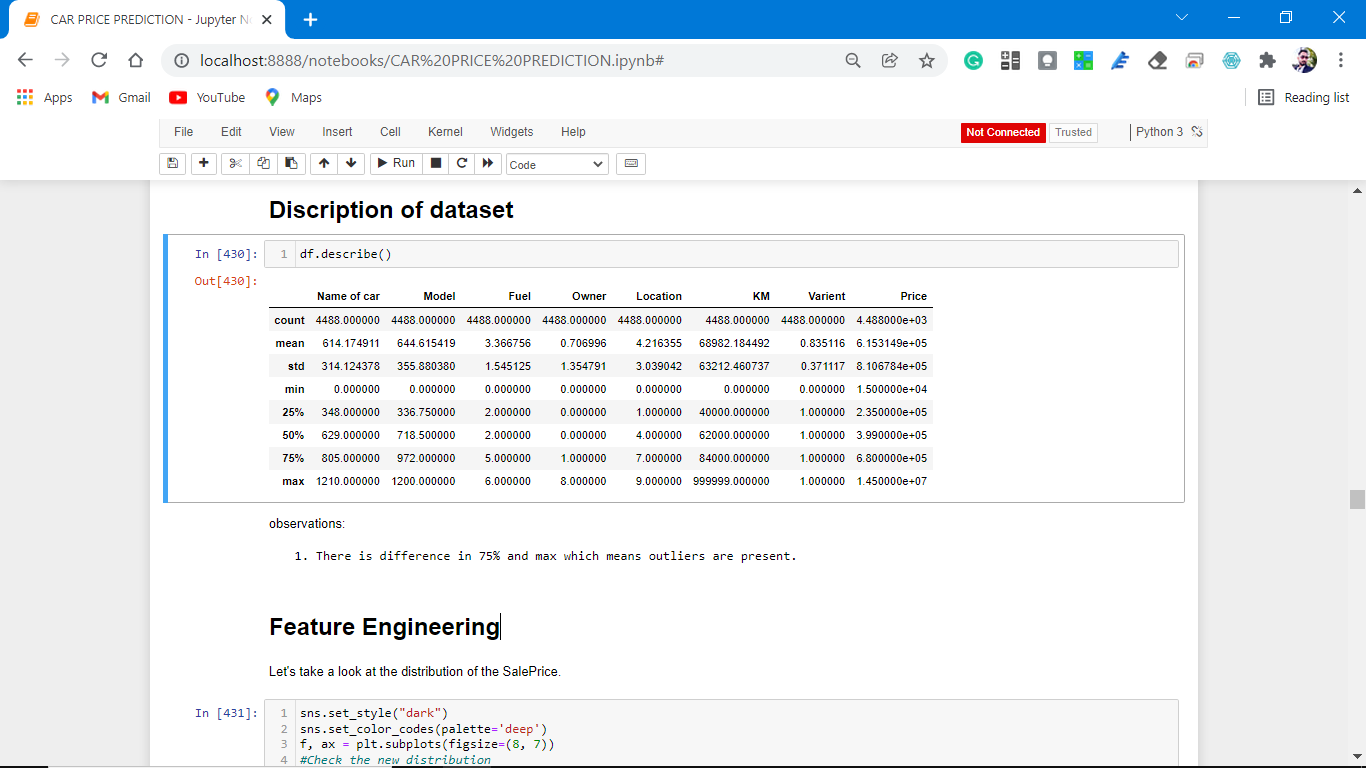
3. when we plot the heat map with notations is true. we can easily observe correlation.

4. All the columns are negatively correlated with the Price column except location.

5. hence no feature column is increasing the price of the car.

5. Some feature columns are negative which impacts on accuracy and learning of the M.L model.

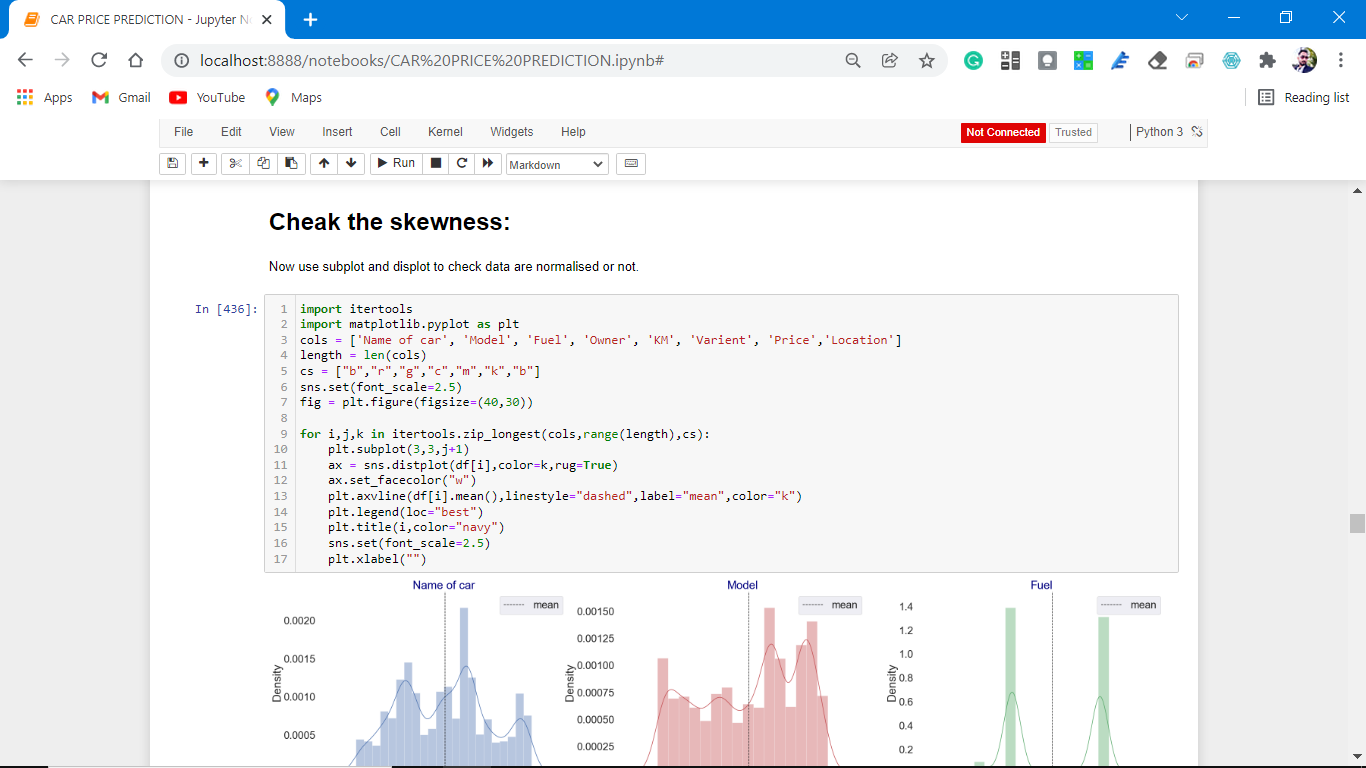
**Description of the dataset.**

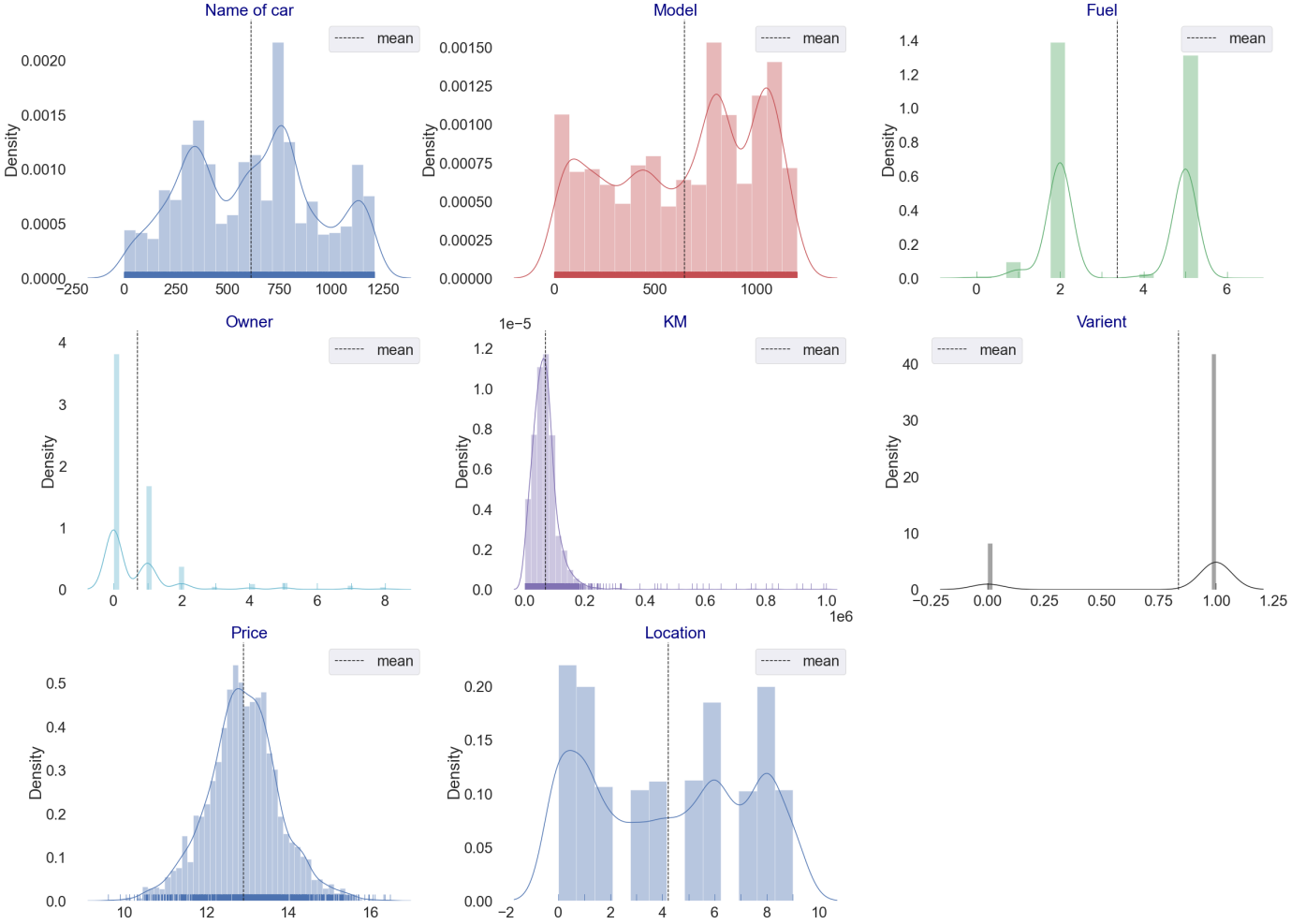
****

observations:

1. There is a difference in 75% and max which means outliers are present.

**Cheak the skewness:**

****

****

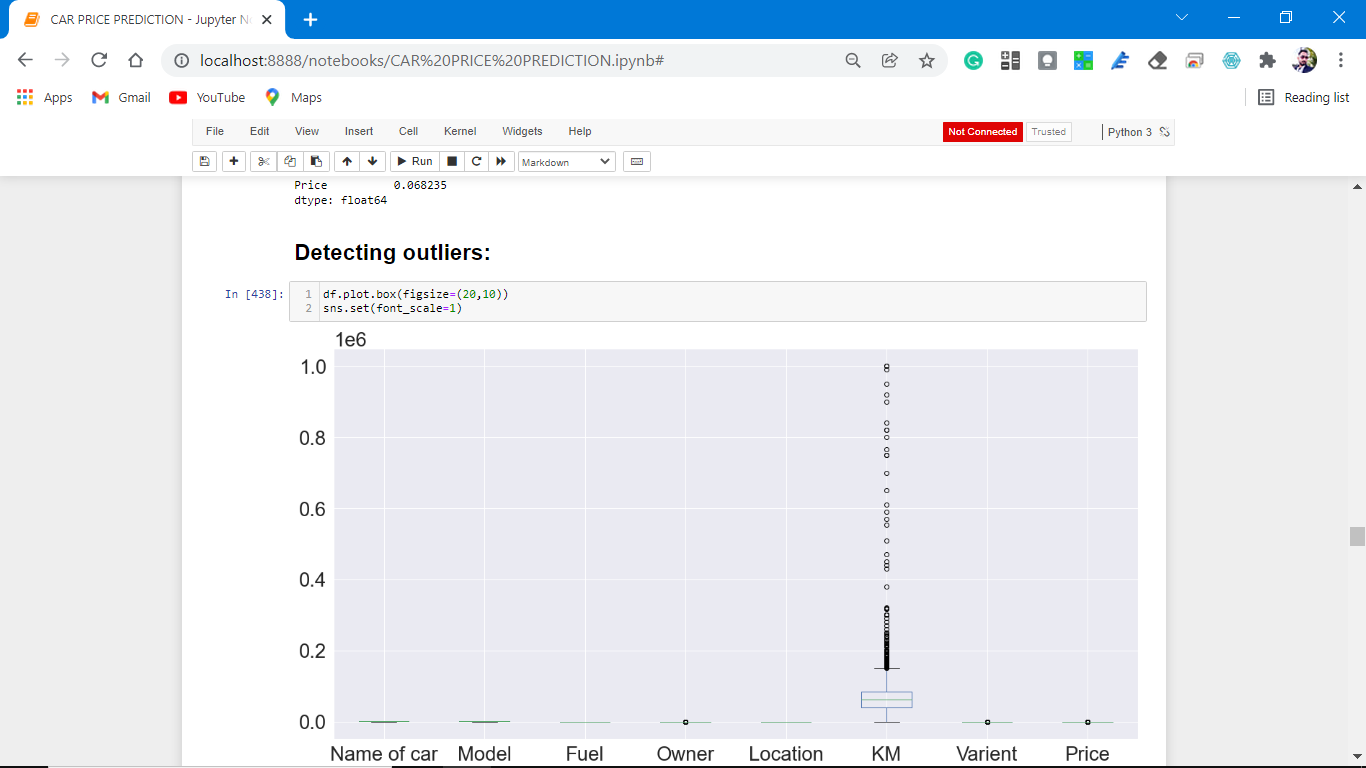
# observations:

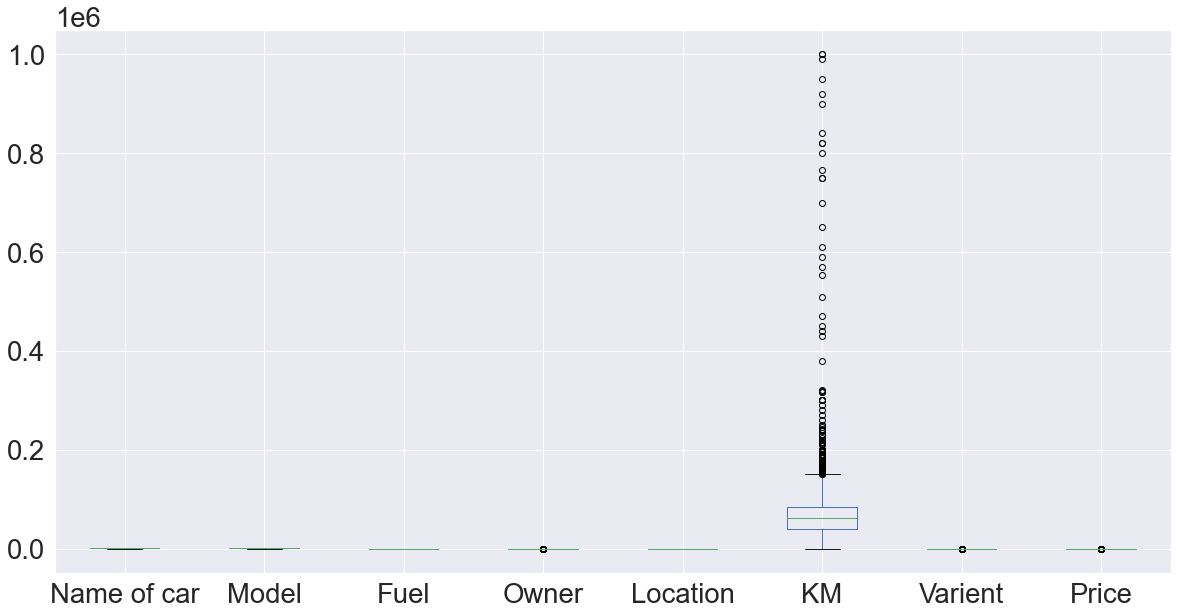
From the above plotting of distribution plot, we see that some features columns are not normally distributed.

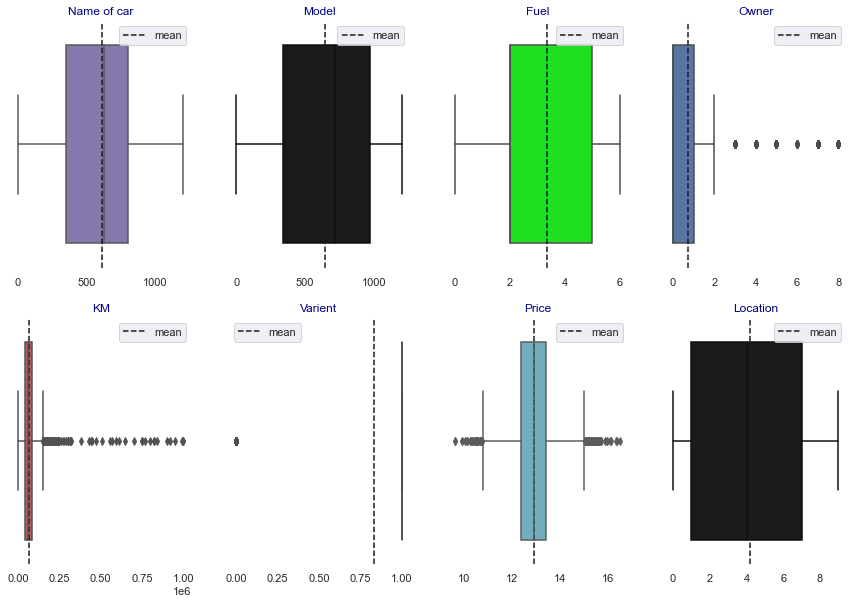
some columns are skewed towards the right.

Building blocks are out of the normal curve hence outliers are present.

**Detecting Outliers**

****

****

****

Observations:

1. From the above plotting we see that outliers are present in the dataset.

2. Most outliers are present in KM and Price column.

**Methods of Removing skewness:**

1. to remove the skewness from the dataset we removed the skewness by using yeo-johnson(A New Family of Power Transformations to Improve Normality of Symmetry).

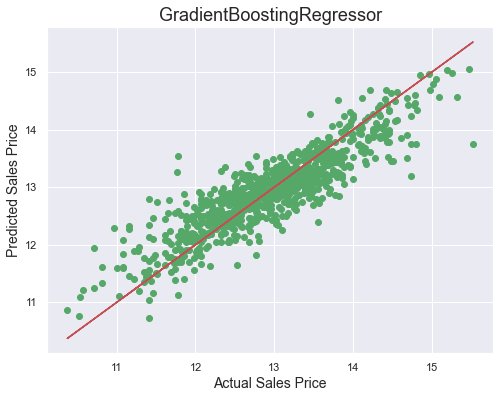
2. After applying power transform our dataset is normalized.

* Interpretation of the Results
* We tested 8 models out of which Gradient Boosting Regressor performed Best as Accuracy score and CV is Optimum.
* The best model is Gradient Boosting Regressor. Since the difference between the percentage score of cross-validation and r2\_score is optimum.

# At cv:- 17

# R2 Score: 77.94710770113188[¶](http://localhost:8888/notebooks/CAR%20PRICE%20PREDICTION.ipynb#R2-Score:-77.94710770113188)

# Cross Val Score: 74.3798966811026



**CONCLUSION**

* Key Findings and Conclusions of the Study

So, our Aim is achieved as we have successfully ticked

all our parameters as mentioned in our Aim Column. It is seen Location is the most effective attribute in predicting the label column and that the Gradient Boosting regression is the most effective model for our Dataset with cv and accuracy is 78%.

* Learning Outcomes of the Study in respect of Data Science

That's it! We reached the end of our exercise.

Throughout this kernel, we put into practice many of the strategies for predicting the resale value of the cars. We philosophized about the variables, we analyzed 'Price' alone and with the most correlated variables, we dealt with missing data and outliers, we tested some of the fundamental statistical assumptions and we even transformed object variables into int variables. That's a lot of work that Python helped us make easier.

* Limitations of this work and Scope for Future Work

Limitations of this work are as follows:

1. This study works well predicting the valuations but is limited when comes to different states and cities as the location has a great impact on predicting the valuations.

For future work, we need data from foreign markets as well or mixed data globally.

Also, the accuracy of predicting is not 100% so many more models to test and which predict with 100% accuracy.

That’s all from this Project Report.

**Thank you**

Data exploration is the first step in data analysis and typically involves summarizing the main

characteristics of a data set, including its size, accuracy, initial patterns in the data, and other

attributes. It is commonly conducted by data analysts using visual analytics tools, but it can

also be done in more advanced statistical software, Python. Before it can analyze

data collected by multiple data sources and stored in data warehouses, an organization must

know how many cases are in a data set, what variables are included, how many missing

values there are, and what general hypotheses the data is likely to support. An initial

exploration of the data set can help answer these questions by familiarizing analysts with the

data with which they are working.

We divided the data 9:1 for Training and Testing purposes respectively.

Data exploration is the first step in data analysis and typically involves summarizing the main

characteristics of a data set, including its size, accuracy, initial patterns in the data, and other

attributes. It is commonly conducted by data analysts using visual analytics tools, but it can

also be done in more advanced statistical software, Python. Before it can analyze

data collected by multiple data sources and stored in data warehouses, an organization must

know how many cases are in a data set, what variables are included, how many missing

values there are, and what general hypotheses the data is likely to support. An initial

exploration of the data set can help answer these questions by familiarizing analysts with the

data with which they are working.

We divided the data 9:1 for Training and Testing purposes respectively.