**Project Report   
on   
Used Cars Price Prediction**

Submitted by:

Saikrishna Padamatinti

**Problem Statement:**

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. One of our clients works with small traders, who sell used cars. With the change in 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 car price valuation model. This project contains two phase.

**Data Collection Phase** You have to scrape at least 5000 used cars data. You can scrape more data as well, it’s up to you. more the data better the model In this section You need to scrape the data of used cars from websites (Olx, cardekho, Cars24 etc.) You need web scraping for this. You have to fetch data for different locations. The number of columns for data doesn’t have limit, it’s up to you and your creativity. Generally, these columns are Brand, model, variant, manufacturing year, driven kilometers, fuel, number of owners, location and at last target variable Price of the car. This data is to give you a hint about important variables in used car model. You can make changes to it, you can add or you can remove some columns, it completely depends on the website from which you are fetching the data. Try to include all types of cars in your data for example- SUV, Sedans, Coupe, minivan, Hatchback.

After collecting the data, you need to build a machine learning model. Before model building do all data pre-processing steps. Try different models with different hyper parameters and select the best model.

**Data Collection Phase:**

Collected the data from carDekho.com for 3 different locations – Hyderabad, Bangalore and Mumbai.

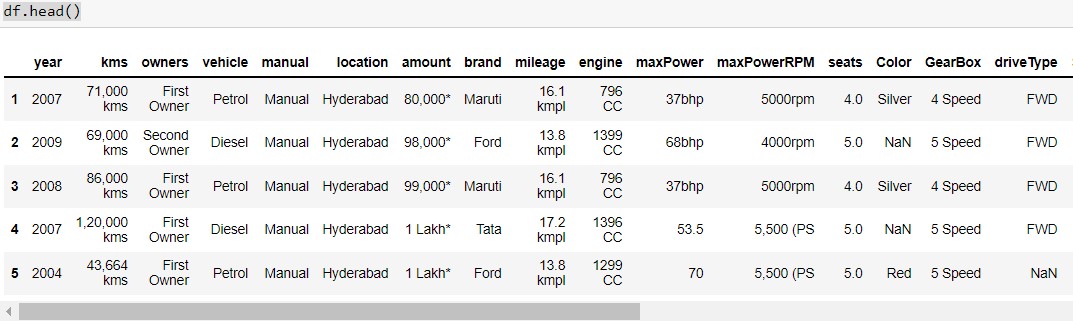
**Exploratory Data Analysis**

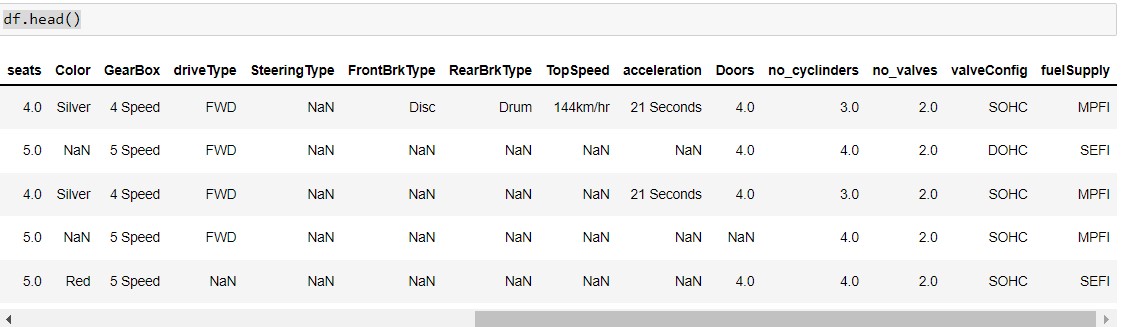
****

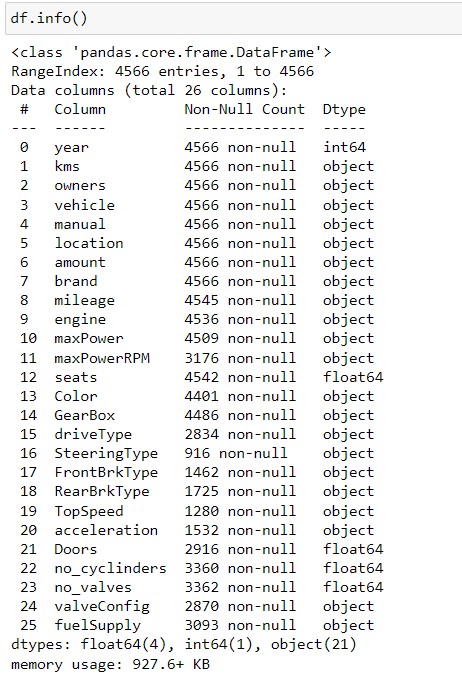
The size of the dataset is 4566 rows with 26 columns.

The target feature “amount” is continuous. Hence it is Regression problem.

To get the overview of the data, looking at first 5 rows.



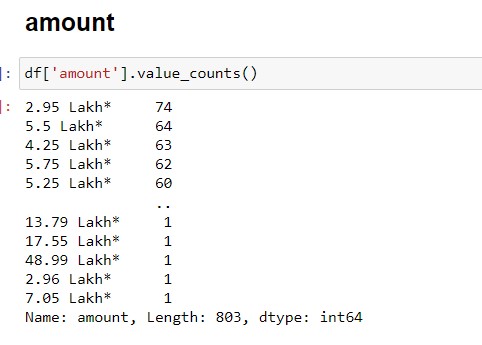




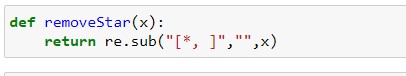
there are missing values in 'mileage', 'engine', 'maxPower', 'maxPowerRPM', 'seats', 'Color', 'GearBox', 'driveType', 'SteeringType', 'FrontBrkType', 'RearBrkType', 'TopSpeed', 'acceleration', 'Doors', 'no\_cyclinders', 'no\_valves', 'valveConfig', 'fuelSupply'.

**Feature Engineering:**

**Amount feature**

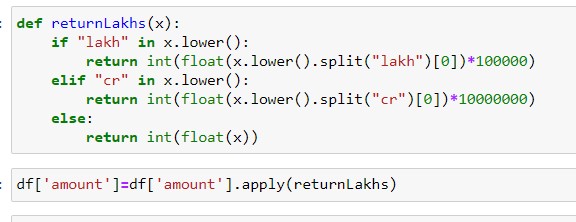
.

Amount feature has extra \* at the end so removed it.

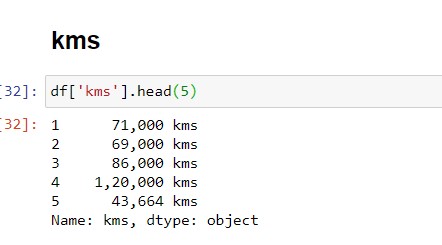


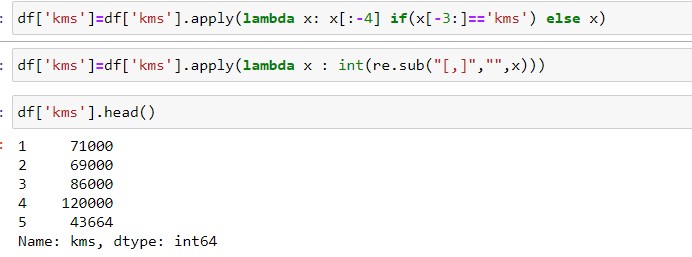


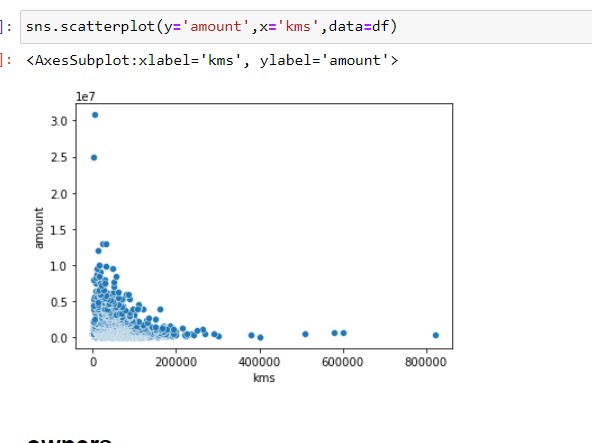
Amount feature has lakh and cr in words, so for lakh multiplying by 100000 and for cr multiplying by 10000000.



**Kilometers feature** has kms at the end of each value. So removing kms at the end. Removing , from the each value and converting it to integer.



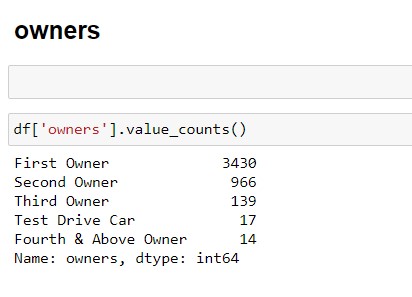




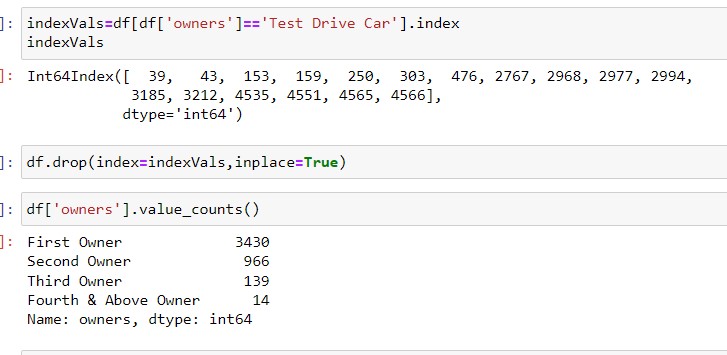
As can be seen from the figure as the kilometers driven increases amount decreases.

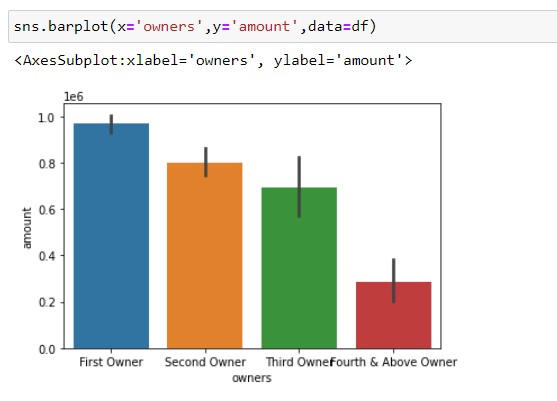
**Owners** feature indicates the number of people the car was used by earlier.

From the data removing the rows with Test Drive Car value.

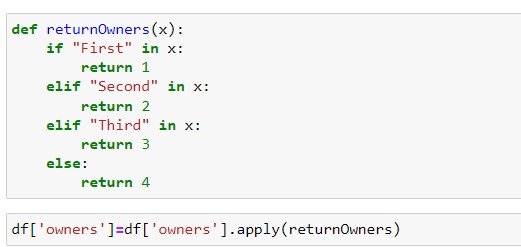


There are 4 unique values in the owners feature which are First owner, second owner, Third owner and Fourth & above owner.





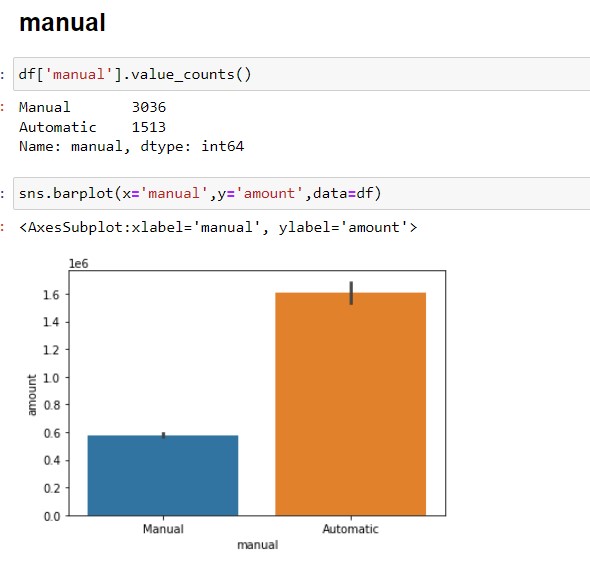
It is evident from the figure that as the number of owners increases the value of the car decreases.



**Vehicle** feature indicates the type of fuel the vehicle runs on. There are five unique values namely Petrol, Diesel, CNG, LPG, Electric.

****

**Manual feature** indicates whether the gear transmission is manual or automatic for the car.

****

There are 3036 cars of manual gear transmission and 1513 cars of automatic gear transmission.

Automatic gear tranmission cars have more price than manual gear transmission cars.

**Location feature** indicates the location the car is sold. There are 2462 cars from Mumbai region, 1340 cars fomr Bangalore region and 747 cars in Hyderabad location.

****

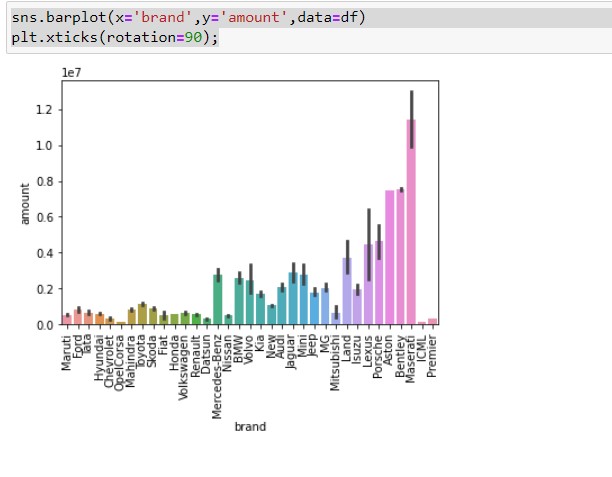
Mean price of Cars sold in bangalore is more than mean price of cars sold in Mumbai and Hyderabad.

**Brand feature**  indicates the brand of the car that is being sold.

There are 35 branded cars in total.

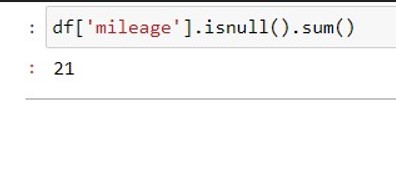


****

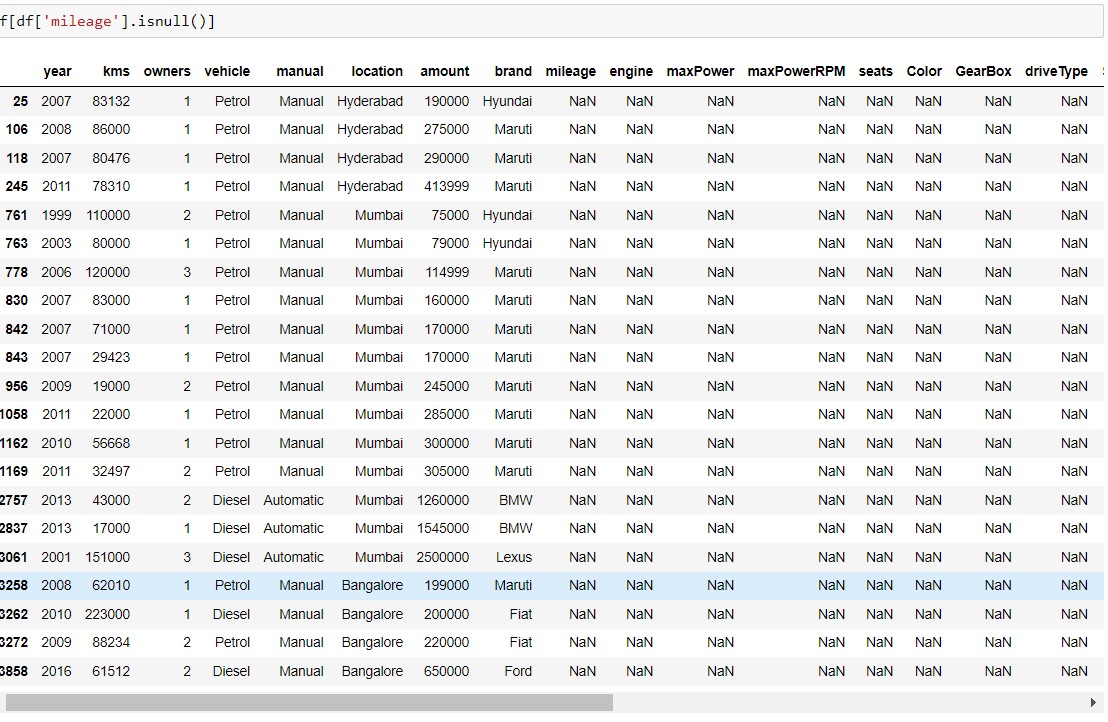
****

Mean price of the Cars of brand Maseratti is more costly than other branded cars.

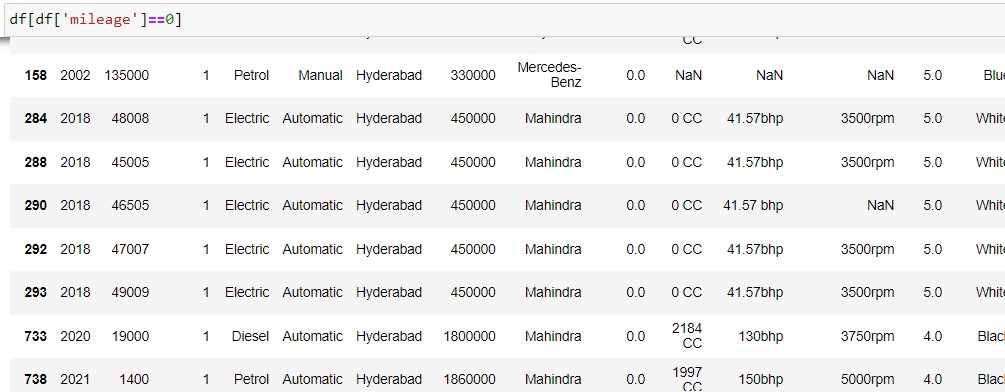
**Mileage:**



There are 21 records with missing values for mileage.



For the records with missing values for mileage other features also does not have values. So dropped all those records with missing values for mileage.

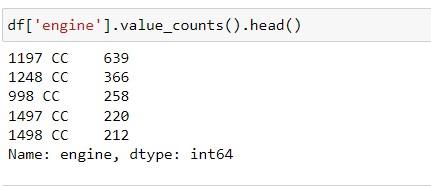


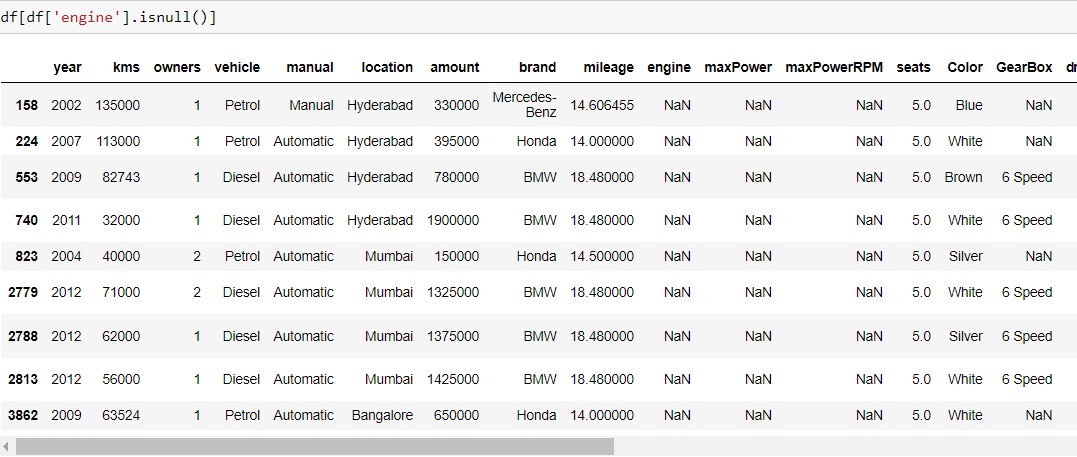
There are some records with 0 value for mileage.



Replaced the 0 values with mean value of mileage after grouping by based on the brand.

**Engine** feature indicates the capacity of the cars.

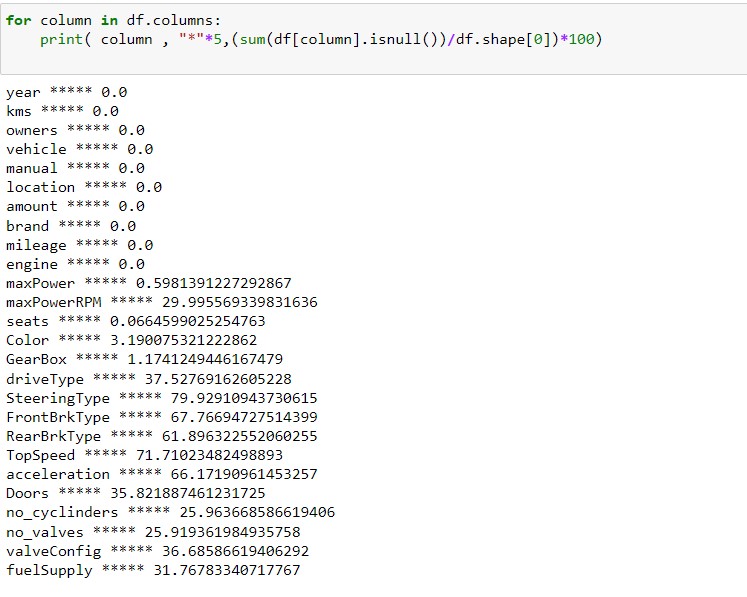




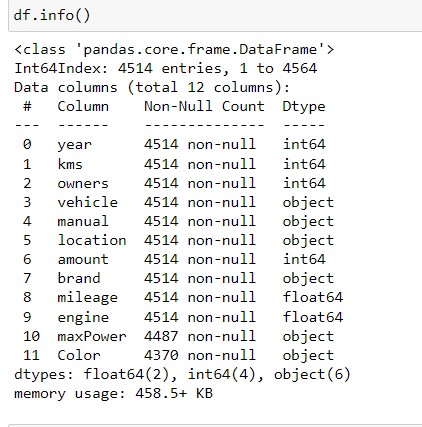


The engine feature has some missing values but for the same records other features also have missing values. So dropped the records with missing values for engine.

**Checking the percentage of missing values for other features.**

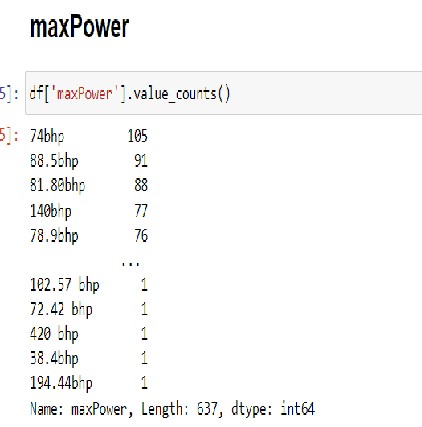


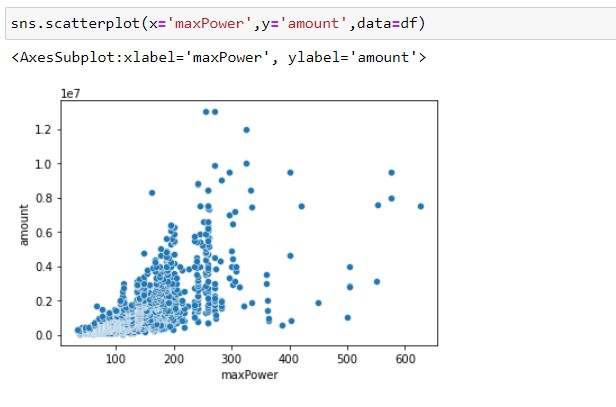
most of the features have more than 20 percent of missing values so dropping those features.



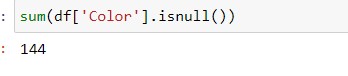
After removing unnecessary features we have 11 independent features and one dependent feature.

**Max Power:** indicates the power of the car.

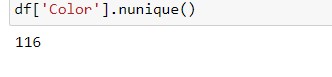




**Color :** indicates the color of the car. There are 144 null values in the car feature.

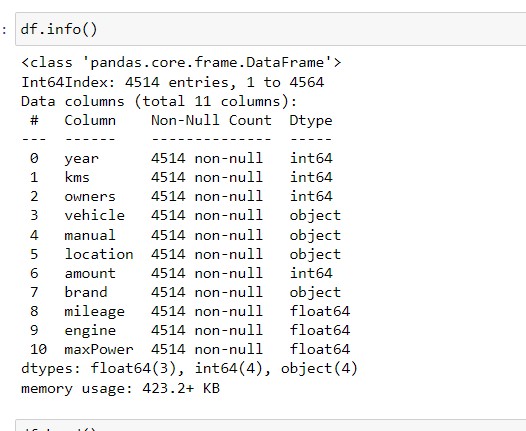


There are 116 unique values in the car feature.



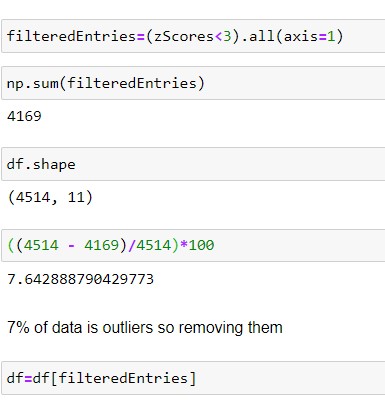
If we do get dummies on the car feature, we get a large number of independent features, so dropping the car feature.

Checking the data types of each feature and number of non null values.

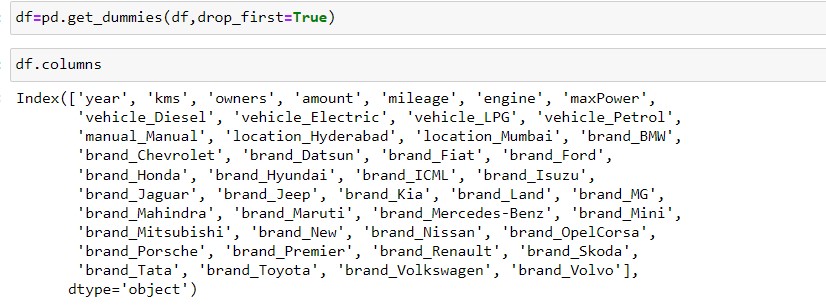


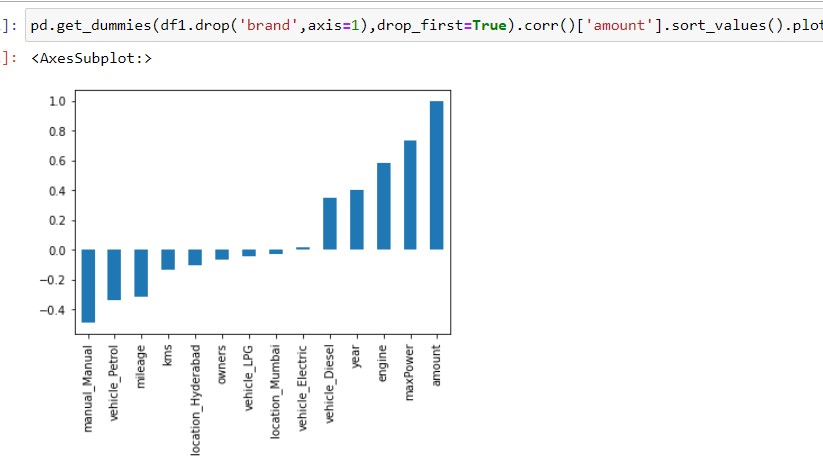
**Checking for outliers using zscore**



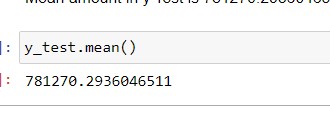


There are 7 percent of outliers in the data and removed them.





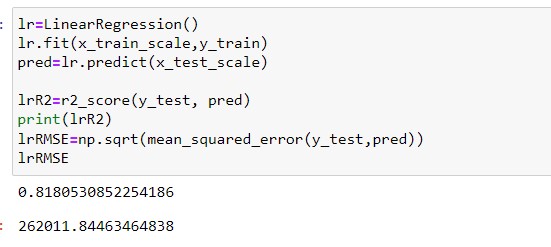
**Model Training:**

****

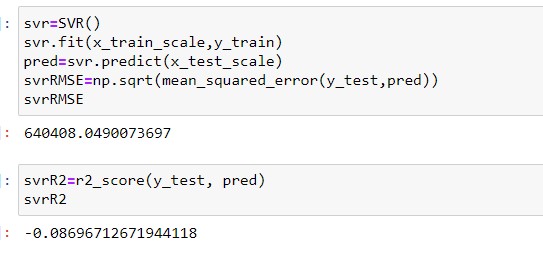
yTest has a mean value of 781270.

**Linear Regression model** fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

**Parameters**

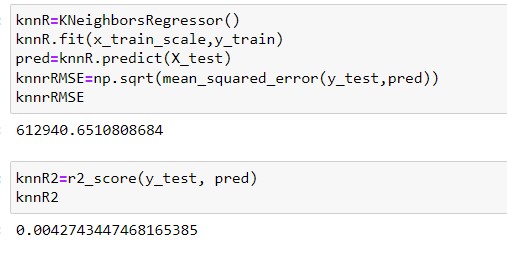
****

**Linear Regression** model has 0.81 r2 score and 262011 Root mean square error.

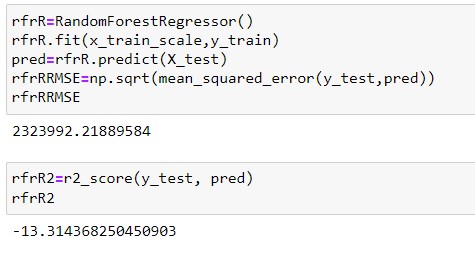
****

**Support Vector Regressor** has a root mean square error value of 640408 and r2 score is negative that mean the model is not able to explain the variance in the data.

**KNeighbors Regressor** The target is predicted by local interpolation of the targets associated of the nearest neighbors in the training set.

****

**K nearest neighbour Regressor** has a root mean squared error value of 6129400 and r2 score of 0.004.

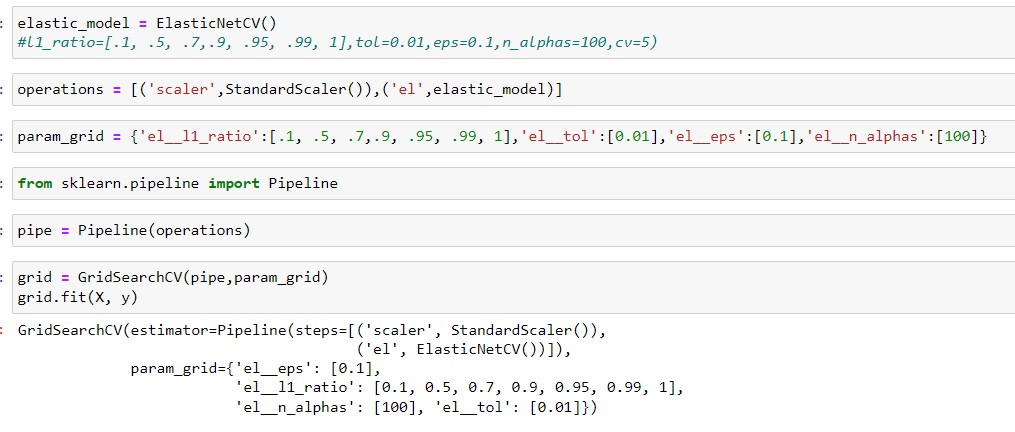
****

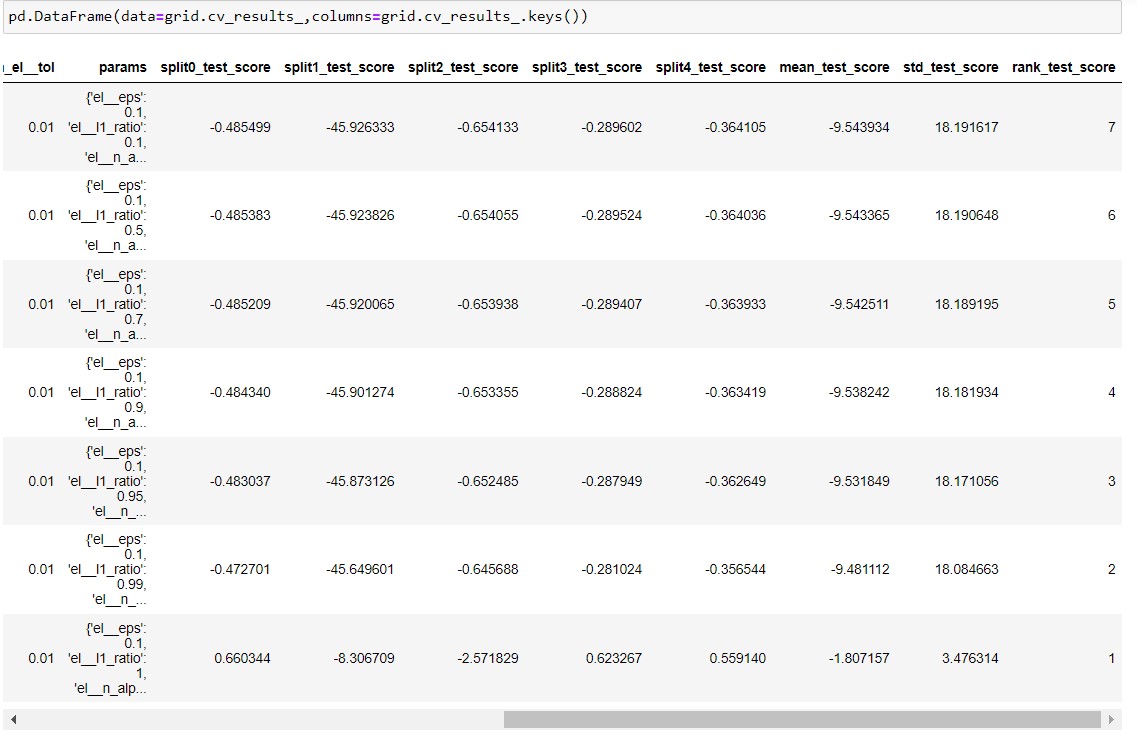
**Random Forest Regressor** is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

Random Forest Regressormodel has a root mean square error of 2323992 very huge compared to other models and r2 score is also very less -13.31

On Comparing the root mean square error values of all models choosing Linear Regression as the best model for HyperParameter Tuning. R2 score is also more for Linear Regression compared to other models.

**Hyper Parameter Tuning – Linear Regression**



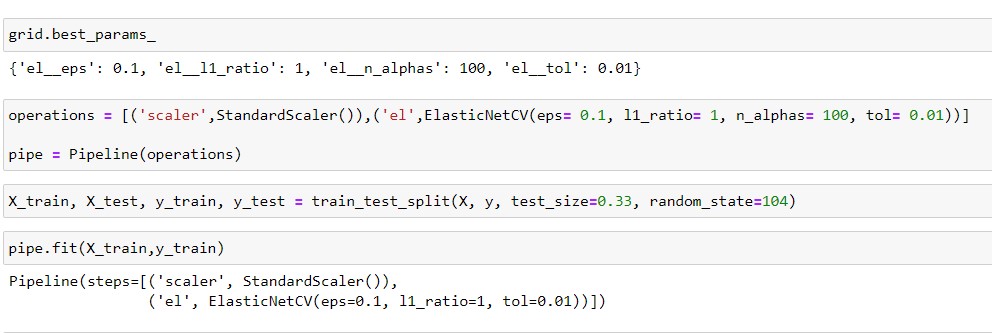


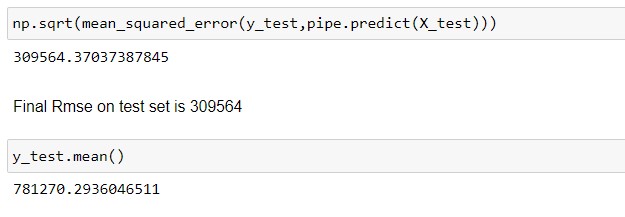
Hyperparameters are the variables that the user specify usually while building the Machine Learning model. Thus, hyperparameters are specified before specifying the parameters or we can say that hyperparameters are used to evaluate optimal parameters of the model. the best part about hyperparameters is that their values are decided by the user who is building the model. For example, max\_depth in Random Forest Algorithms, k in KNN Classifier.

Grid Search uses a different combination of all the specified hyperparameters and their values and calculates the performance for each combination and selects the best value for the hyperparameters. This makes the processing time-consuming and expensive based on the number of hyperparameters involved.

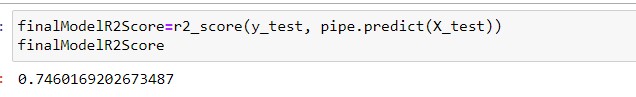
In GridSearchCV, along with Grid Search, cross-validation is also performed. Cross-Validation is used while training the model. As we know that before training the model with data, we divide the data into two parts – train data and test data. In cross-validation, the process divides the train data further into two parts – the train data and the validation data.

**Training the final model with the best parameters based on GridSearchCV:**

Creating a pipeline with scaler – fitting the scaler on training data, transforming the training and testing data, elasticNet model – for prediction. ****



After training the model on the training data with the best parameters, final root mean squared error achieved is 309564 and r2 score achieved is 0.74.



In Conclusion, the ElasticNet is chosen as the best model with Root mean squared error of 309564.