ISSN No:-2456-2165

Prediction of Resale Value of the Car Using Linear Regression Algorithm

Kiran S
Computer Science Engineering
SJB Institute of Technology
Bangalore, India

Abstract:- The expected estimate for resale value of a car is most significant in the field of present research and technology. Most significant attributes are considered for predicting the resale value of the car. The significant relationships among various attributes are found by establishing the correlations. In this research the price of the car is considered as dependent variable for target prediction .The data used for prediction was taken from web. The suitability of linear regression algorithm is identified and implemented in this research work for accurately predicting the resale value of the vehicle based on most significant attributes that are been selected on the basis of highest correlation. The outcome of the research shows that the accuracy of the model built is upto 90 percent and error obtained is 10 percent.

Keywords:- *Machine Learning , Prediction, Linear regression, Target , Correlation*

I. INTRODUCTION

Predicting the resale value of a car is not a easy job to be performed. It requires adequate knowledge because the value of used cars depends on a number of factors. The most significant ones are usually the age of the car, its make and model, the origin of the car (the original country of the manufacturer), its mileage (the number of kilometres it has run) and its horsepower. Due to increase in fuel prices, fuel economy is also of greater importance. Practically speaking, many people do not know exactly how many kilometres their cars can be driven for each litre of either petrol or diesel. Different factors for example the type of fuel it uses, the interior and exterior style, the kind of breaking system, acceleration, the volume of its cylinders (measured in cc), safety features, its size, number of seating capacity, total number of doors, body colour of car, weight of the car, consumer reviews, prestigious awards won by the car manufacturer, its physical built quality, is it a sports car, if it has cruise control, is it automatic or manual transmission, is it owned by an individual or a company and other options such as air conditioner and purifiers, sound system with audio and video controls, power steering, alloy wheels, GPS navigator all may change the price of car . Some special things which buyers attach for importance in Mauritius is the car local or previous owners, if the car had been involved in serious accidents and if it is a lady-driven car. The look and feel of the car is most important factors for the price of the car. It is noticed that price probably depends on a huge number of factors. In reality, the information about all these factors is not always available and the buyers are forced to make the decision to purchase without clear information provided. In the present times it is observed that the prediction of used car for second hand cars are done manually and it is based on some basic parameters. Cars 24 is best example wherein there is a manual testing done for car in order to know the resale value of the car. So it takes huge amount of time and space to manually predict the price of the car .Taking into consideration of all the factors mentioned above, this paper is built where the prediction of resale value of the car is done automatically by the model built and it is totally computerized. This research makes use of Linear Regression Algorithm from Machine Learning for prediction of car price.

II. LITERATURE SURVEY

Doan Van Thai ; Luong Ngoc Son ; Pham Vu Tien ; Nguyen Nhat Anh ; Nguyen Thi Ngoc Anh[1] This paper will include the procedures for extraction of importance, information derivation, and rules for subjective information. The fundamental reason for the flow research is to investigate various information kinds of vehicle information and the goal is to make a mechanized procedure to foresee vehicle costs.

Nitis Monburinon ; Prajak Chertchom ; Thongchai Kaewkiriya ; Suwat Rungpheung ; Sabir Buya ; Pitchayakit Boonpou[2] report on execution of relapse dependent on directed ML models. Each model is prepared utilizing information of trade-in vehicle showcase gathered from German web based business site. Thus, inclination helped relapse trees gives the best execution with mean total mistake (MSE) =3D 0.28. . Followed by arbitrary woodland relapse with MSE =3D 0.35 and various direct relapse with MSE =3D 0.55 separately.

Ning Sun ; Hongxi Bai ; Yuxia Geng ; Huizhu Shi[3] In this paper, the value assessment model dependent on enormous information investigation is proposed, which exploits generally circled vehicle information and countless vehicle exchange information to dissect the value information for each sort of vehicles by utilizing the improved BP neural system calculation. It intends to set up a recycled vehicle value assessment model to get the value that best matches the vehicle.

III. OBJECTIVES

- To find the correlation with each attribute to that of target attribute
- To draw a linear regression curve with the target attribute
- Finding out the total error and accuracy

IV. METHODOLOGY

Linear 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. Different regression models differ based on – the kind of relationship between dependent and independent variables, they are considering and the number of independent variables being used.

Software Specifications: Anaconda Navigator, Jupiter notebook, Machine learning packages: Sklearn, Matplotlib, Python ADE, Microsoft window 2010.

V. IMPLEMENTATION AND ANALYSIS

Machine Learning algorithms can be implemented on some standard data having no redundancy. we mainly require the raw data from some standard sources that are available in the web. Initially data can be in either of two forms i.e. structured data or unstructured data. Structured data is normally in the form of rows and columns, it can be a database or a dataset. whereas unstructured data is in the form of images , videos, audio tapes etc. The flow of execution of the work done is shown in Figure 1

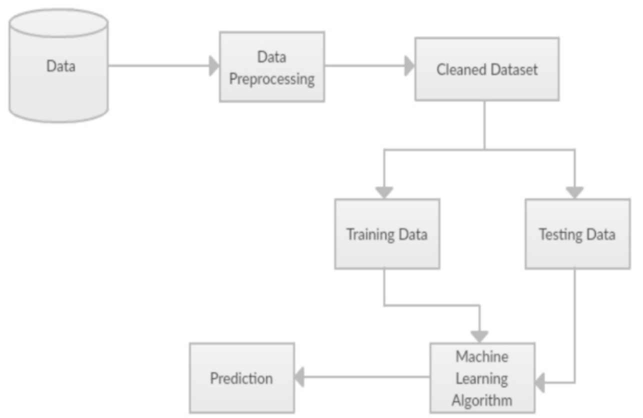
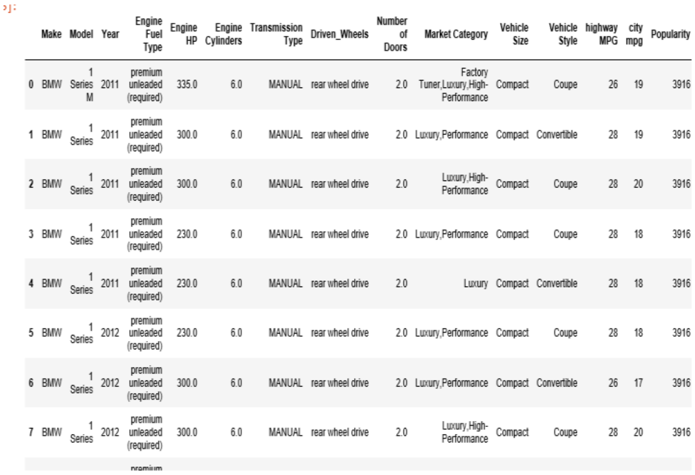


Fig 1:- Flow of Execution

Initially the data is collected which is in the form of dataset. The data present in dataset is then pre-processed with some data mining technique. Once the data is pre processed then cleansing of the data is done in order to reduce the redundancy once the data is perfectly ready , then the data instances are divided into training data and testing data . This data is then fed into a machine learning algorithm and thus prediction of target attribute is done.

VI. RESULTS AND DISCUSSIONS

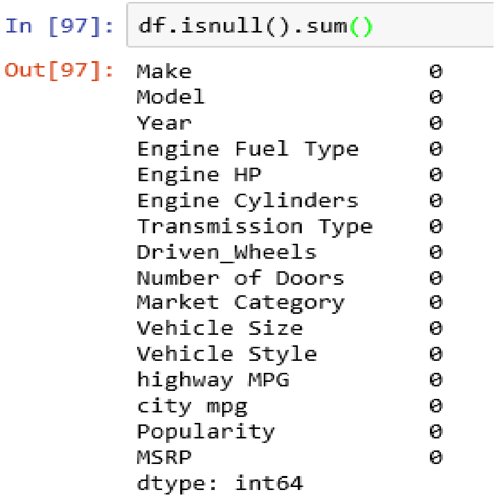
Initially the dataset is used from an standard source from web. This dataset which is in the form of rows and columns is initially loaded into the Jupiter notebook. This dataset is mainly required to predict the target attribute. The dataset is imported into the code by the command car price=pd.read_csv("data.csv") where in the dataset is in the format of csv file and the dataset is given with a variable called car price.



	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	city mpg	Popularity
0	BMW	1 Series M	2011	premium unleaded (required)	335.0	6.0	MANUAL	rear wheel drive	2.0	Factory Tuner,Luxury,High-Performance	Compact	Coupe	26	19	3916
1	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Convertible	28	19	3916
2	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,High-Performance	Compact	Coupe	28	20	3916
3	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Coupe	28	18	3916
4	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury	Compact	Convertible	28	18	3916
5	BMW	1 Series	2012	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Coupe	28	18	3916
6	BMW	1 Series	2012	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Convertible	26	17	3916
7	BMW	1 Series	2012	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,High-Performance	Compact	Coupe	28	20	3916

Fig 2:- Dataset Used

The Fig 2 shows the dataset used which contains 16 attributes and 14 thousand instances. All 16 attributes are Make, Model, Year, Engine fuel type, engine hp, engine cylinders, transmission type , driven wheels, number of doors, market category, vehicle size, vehicle style , highway mpg, city mpg, popularity, msrp. In this dataset msrp is the target tribute that is predicted by using all other 15 attributes.

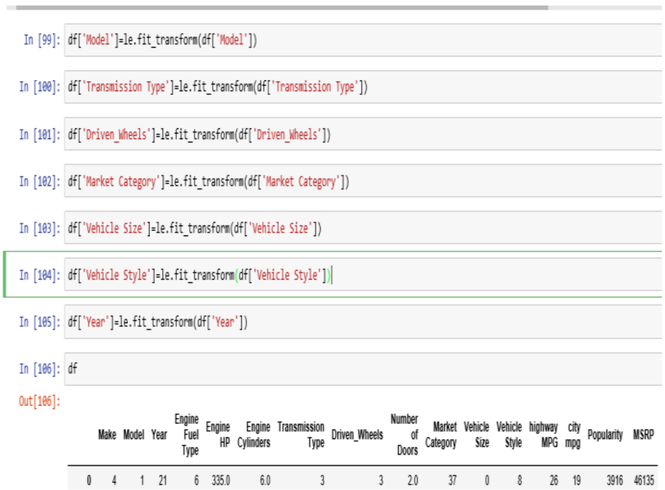


```
In [97]:  df.isnull().sum()

Out[97]:  Make                 0
          Model                0
          Year                 0
          Engine Fuel Type     0
          Engine HP            0
          Engine Cylinders     0
          Transmission Type    0
          Driven_Wheels        0
          Number of Doors      0
          Market Category      0
          Vehicle Size         0
          Vehicle Style        0
          highway MPG          0
          city mpg             0
          Popularity           0
          MSRP                 0
          dtype: int64
```

Fig 3:- Finding Out Null Values

The Fig 3 shows the null values present in each attributes This can be achieved by the command df.isnull( ).sum( ) where in df is the variable declared for dataframe and isnull( ) is the function to check whether there are any null values and sum( ) is to add up and return all those missing values present in each attribute. The result in the

Fig 3 shows that , there are no null values present in each of the attributes in the dataset. So there is no necessity so removing and cleaning any null values for this dataset.

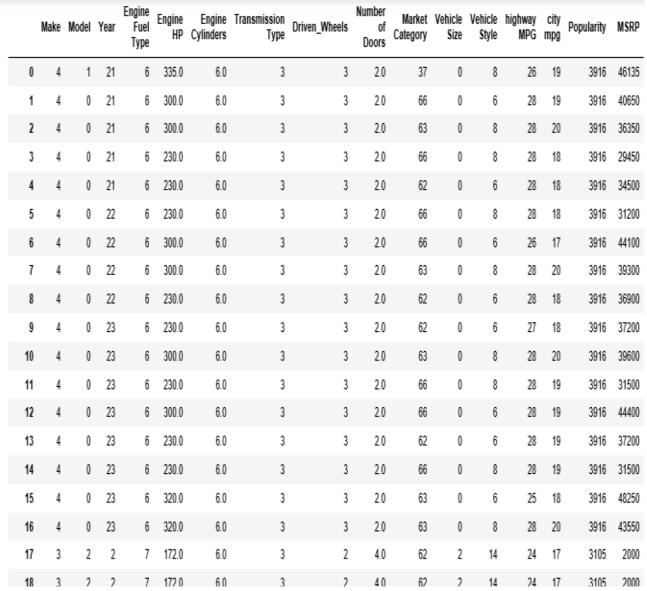


```
In [99]:  df['Model']=le.fit_transform(df['Model'])

In [100]: df['Transmission Type']=le.fit_transform(df['Transmission Type'])

In [101]: df['Driven_Wheels']=le.fit_transform(df['Driven_Wheels'])

In [102]: df['Market Category']=le.fit_transform(df['Market Category'])

In [103]: df['Vehicle Size']=le.fit_transform(df['Vehicle Size'])

In [104]: df['Vehicle Style']=le.fit_transform(df['Vehicle Style'])

In [105]: df['Year']=le.fit_transform(df['Year'])

In [106]: df

Out[106]:
```

	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	city mpg	Popularity	MSRP
0	4	1	21	6	335.0	6.0	3	3	2.0	37	0	8	26	19	3916	46135

Fig 4:- Converting String into Integers

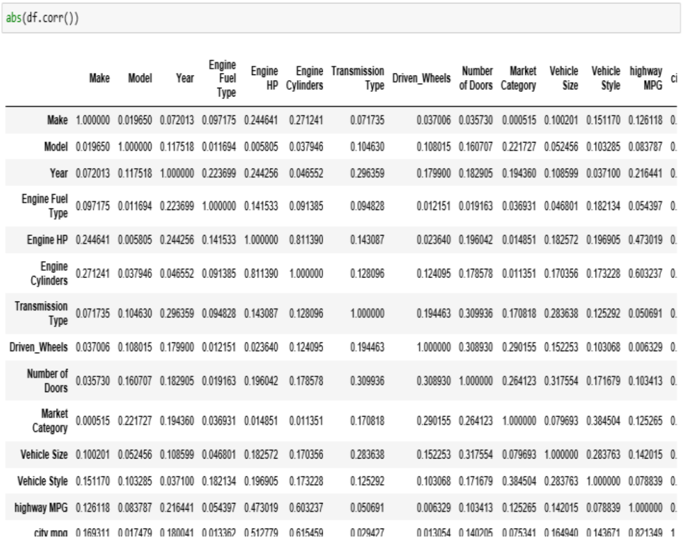
The Fig 4 shows the conversion of strings to integers . Dataset contains values for attributes in strings as well as in integers. But strings cannot be compared to integers in Machine Learning models. So strings to be converted to integers or integers to be converted into strings. Here Strings are converted into integers by a function called fit_transform( ) which converts strings to integers. All the attribute names should be passed into this function separately so as to convert all the values in the particular attribute.



	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	city mpg	Popularity	MSRP
0	4	1	21	6	335.0	6.0	3	3	2.0	37	0	8	26	19	3916	46135
1	4	0	21	6	300.0	6.0	3	3	2.0	66	0	6	28	19	3916	40650
2	4	0	21	6	300.0	6.0	3	3	2.0	63	0	8	28	20	3916	36350
3	4	0	21	6	230.0	6.0	3	3	2.0	66	0	8	28	18	3916	29450
4	4	0	21	6	230.0	6.0	3	3	2.0	62	0	6	28	18	3916	34500
5	4	0	22	6	230.0	6.0	3	3	2.0	66	0	8	28	18	3916	31200
6	4	0	22	6	300.0	6.0	3	3	2.0	66	0	6	26	17	3916	44100
7	4	0	22	6	300.0	6.0	3	3	2.0	63	0	8	28	20	3916	39300
8	4	0	22	6	230.0	6.0	3	3	2.0	62	0	6	28	18	3916	36900
9	4	0	23	6	230.0	6.0	3	3	2.0	62	0	6	27	18	3916	37200
10	4	0	23	6	300.0	6.0	3	3	2.0	63	0	8	28	20	3916	39600
11	4	0	23	6	230.0	6.0	3	3	2.0	66	0	8	28	19	3916	31500
12	4	0	23	6	300.0	6.0	3	3	2.0	66	0	6	28	19	3916	44400
13	4	0	23	6	230.0	6.0	3	3	2.0	62	0	6	28	19	3916	37200
14	4	0	23	6	230.0	6.0	3	3	2.0	66	0	8	28	19	3916	31500
15	4	0	23	6	320.0	6.0	3	3	2.0	63	0	6	25	18	3916	48250
16	4	0	23	6	320.0	6.0	3	3	2.0	63	0	8	28	20	3916	43550
17	3	2	2	7	172.0	6.0	3	2	4.0	62	2	14	24	17	3105	2000
18	3	2	2	7	172.0	6.0	3	2	4.0	62	2	14	24	17	3105	2000

Fig 5:- Dataset Converted into Integers

The Fig 5 shows the results of string to integer conversion as mentioned in Fig 4. So in the fig 5 the total dataset has been converted into integers. Now it turns out to be easy to compare the attributes.

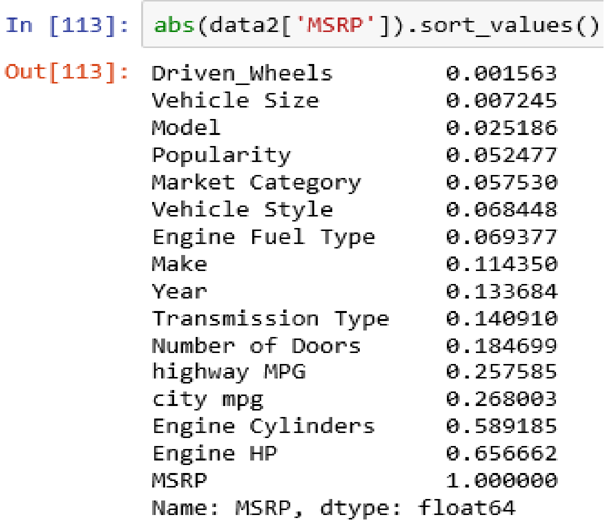


```
abs(df.corr())
```

	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	ci
Make	1.000000	0.019650	0.072013	0.097175	0.244641	0.271241	0.071735	0.037006	0.035730	0.000515	0.100201	0.151170	0.126118	0.
Model	0.019650	1.000000	0.117518	0.011694	0.005805	0.037946	0.104630	0.108015	0.160707	0.221727	0.052456	0.103285	0.083787	0.
Year	0.072013	0.117518	1.000000	0.223699	0.244256	0.046552	0.296359	0.179900	0.182905	0.194360	0.108599	0.037100	0.216441	0.
Engine Fuel Type	0.097175	0.011694	0.223699	1.000000	0.141533	0.091385	0.094828	0.012151	0.019163	0.036931	0.046801	0.182134	0.054397	0.
Engine HP	0.244641	0.005805	0.244256	0.141533	1.000000	0.811390	0.143087	0.023640	0.196042	0.014851	0.182572	0.196905	0.473019	0.
Engine Cylinders	0.271241	0.037946	0.046552	0.091385	0.811390	1.000000	0.128096	0.124095	0.178578	0.011351	0.170356	0.173228	0.603237	0.
Transmission Type	0.071735	0.104630	0.296359	0.094828	0.143087	0.128096	1.000000	0.194463	0.309936	0.170818	0.283638	0.125292	0.050691	0.
Driven_Wheels	0.037006	0.108015	0.179900	0.012151	0.023640	0.124095	0.194463	1.000000	0.308930	0.290155	0.152253	0.103068	0.006329	0.
Number of Doors	0.035730	0.160707	0.182905	0.019163	0.196042	0.178578	0.309936	0.308930	1.000000	0.264123	0.317554	0.171679	0.103413	0.
Market Category	0.000515	0.221727	0.194360	0.036931	0.014851	0.011351	0.170818	0.290155	0.264123	1.000000	0.079693	0.384504	0.125265	0.
Vehicle Size	0.100201	0.052456	0.108599	0.046801	0.182572	0.170356	0.283638	0.152253	0.317554	0.079693	1.000000	0.283763	0.142015	0.
Vehicle Style	0.151170	0.103285	0.037100	0.182134	0.196905	0.173228	0.125292	0.103068	0.171679	0.384504	0.283763	1.000000	0.078839	0.
highway MPG	0.126118	0.083787	0.216441	0.054397	0.473019	0.603237	0.050691	0.006329	0.103413	0.125265	0.142015	0.078839	1.000000	0.
city mpg	0.169311	0.017479	0.180041	0.013362	0.512779	0.615459	0.029427	0.013054	0.140205	0.075341	0.164940	0.143671	0.821349	1

Fig 6:- Finding out Correlation Between all the Attributes

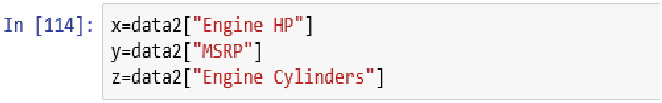
The Fig 6 shows the correlation values between all the attributes that correlation between each and every attrite is found out by the function corr( ) . The max value of correlation is 1 and minimum value is 0 . The command abs(df.corr( )) defines that absolute of correlation is found out so that if there are any negative correlations , then it can be converted into positive . This Absolute of correlation is done so as to keep all the correlation vales between 0 to 1.



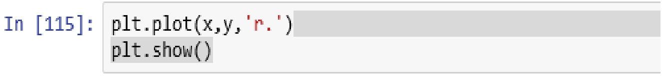
```
In [113]:  abs(data2['MSRP']).sort_values()

Out[113]:  Driven_Wheels          0.001563
           Vehicle Size           0.007245
           Model                  0.025186
           Popularity             0.052477
           Market Category        0.057530
           Vehicle Style          0.068448
           Engine Fuel Type       0.069377
           Make                   0.114350
           Year                   0.133684
           Transmission Type      0.140910
           Number of Doors        0.184699
           highway MPG            0.257585
           city mpg               0.268003
           Engine Cylinders       0.589185
           Engine HP              0.656662
           MSRP                   1.000000
           Name: MSRP, dtype: float64
```

Fig 7:- Sorting all Correlations in Ascending Order

The Fig 7 shows the sorted correlation values of all the attributes to that of target attribute msrp. The highest correlated attributes to that of target attribute msrp are Engine hp and Engine cylinders having 0.65 and 0.58 correlation values. Therefore Engine HP and Engine cylinders are taken for consideration to predict the target attribute msrp.



```
In [114]: x=data2["Engine HP"]
          y=data2["MSRP"]
          z=data2["Engine Cylinders"]
```



```
In [115]: plt.plot(x,y,'r.')
          plt.show()
```

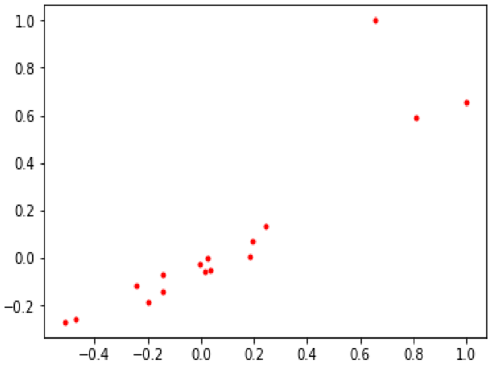
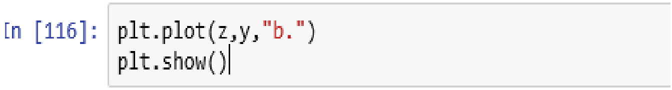


Fig 8:- Plotting Graph between Engine HP v/s MSRP

The Fig 8 shows the plotting of a graph between engine hp and the target msrp. Engine hp is in x axis and target msrp is in y axis. Any attribute that is predicted should always be in y axis according to linear regression model. The values plotted are in red color .



```
In [116]: plt.plot(z,y,"b.")
          plt.show()
```

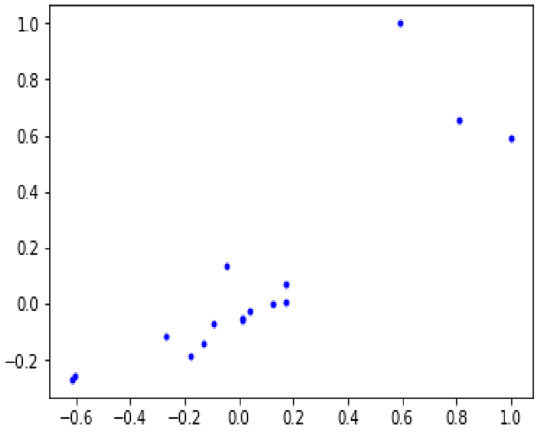
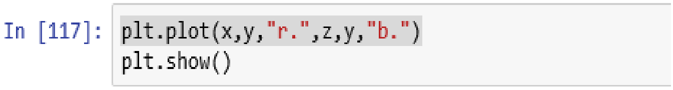


Fig 9:- Plotting of Engine Cylinders V/S Target Attribute MSRP

The Fig 9 shows the plotting of the graph between number of cylinders and msrp. Number of cylinders are in x axis and the target msrp is in y axis . The values plotted are in blue color.



```
In [117]: plt.plot(x,y,"r.",z,y,"b.")
          plt.show()
```

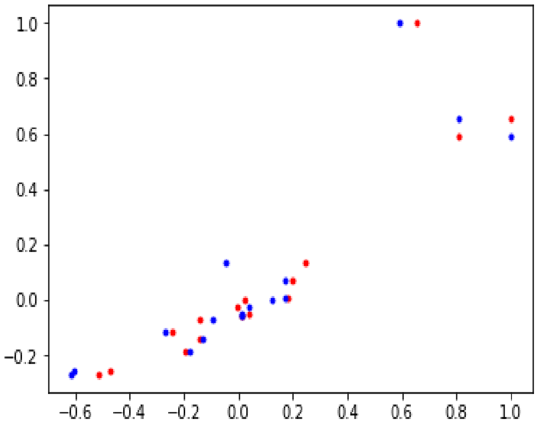
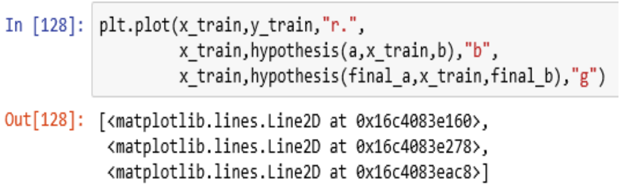


Fig 10:- Plotting of both combinations

The Fig 10 shows plotting of the graphs between engine hp v/s msrp and number of cylinders v/s msrp. Here both engine hp and number of cylinders are plotted in x axis and target msrp is plotted in y axis.



```
In [128]: plt.plot(x_train,y_train,"r.",
                   x_train,hypothesis(a,x_train,b),"b",
                   x_train,hypothesis(final_a,x_train,final_b),"g")

Out[128]: [<matplotlib.lines.Line2D at 0x16c4083e160>,
           <matplotlib.lines.Line2D at 0x16c4083e278>,
           <matplotlib.lines.Line2D at 0x16c4083eac8>]
```

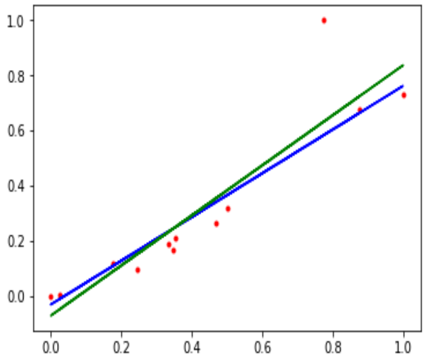
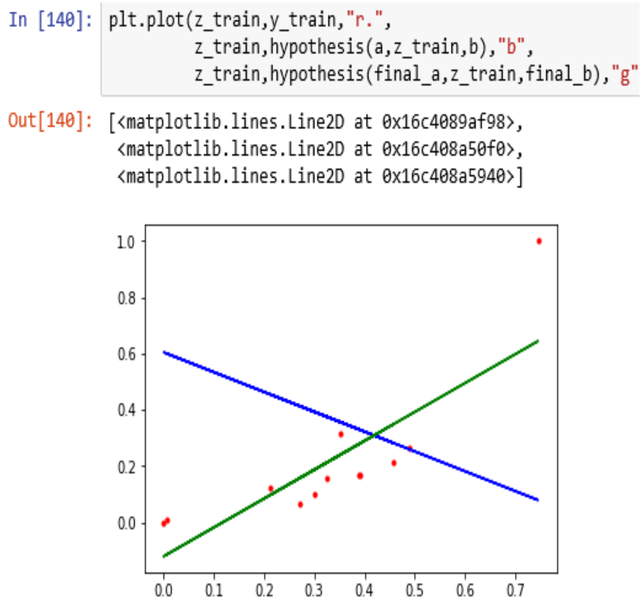


Fig 11:- Best Fit Curve for Engine Hp vs MSRP

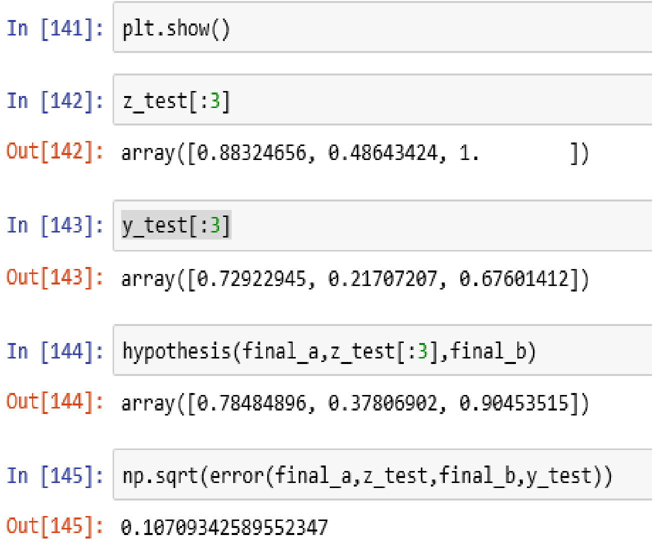
The Fig 11 shows the best fit linear curve between engine hp and msrp . This is the best fit linear curve that is obtained for the values of engine hp and msrp. The investigation shows that blue color curve shows good result than the green curve.



```
In [140]: plt.plot(z_train,y_train,"r.",
                   z_train,hypothesis(a,z_train,b),"b",
                   z_train,hypothesis(final_a,z_train,final_b),"g"

Out[140]: [<matplotlib.lines.Line2D at 0x16c4089af98>,
           <matplotlib.lines.Line2D at 0x16c408a50f0>,
           <matplotlib.lines.Line2D at 0x16c408a5940>]
```

Fig 12:- Best Fit Curve for Engine Cylinders vs MSRP

The Fig 12 shows the best fit linear curve between number of cylinders and msrp . This is the best fit linear curve that is obtained for the values of number of cylinders and msrp. The investigation shows that green color curve shows good result than the blue curve.



```
In [141]: plt.show()

In [142]: z_test[:3]

Out[142]: array([0.88324656, 0.48643424, 1.        ])

In [143]: y_test[:3]

Out[143]: array([0.72922945, 0.21707207, 0.67601412])

In [144]: hypothesis(final_a,z_test[:3],final_b)

Out[144]: array([0.78484896, 0.37806902, 0.90453515])

In [145]: np.sqrt(error(final_a,z_test,final_b,y_test))

Out[145]: 0.10709342589552347
```

Fig 13:- Total Error Percentage

The Fig 13 shows the total error that is obtained by this model. It shows an error of 10.7% with accuracy rate of around 90% . From this investigation we can note that linear regression model in machine learning is providing good efficiency and it performing well with the dataset that is used.

VII. CONCLUSIONS

The Linear Regression model for prediction of resale value of the car is providing an accuracy of 90% . Linear Regression Model is giving an error of 10%. Linear Regression model is better suited for prediction of target attribute that is msrp (car price) and it is performing very good . Further this work can be implemented using different machine learning algorithms and approaches in order to get higher accuracy rate and lower error percentage.

REFERENCES

[1]. Doan Van Thai , Luong Ngoc Son , Pham Vu Tien , Nguyen Nhat Anh and Nguyen Thi Ngoc Anh, “ Prediction car prices using quantify qualitative data and knowledge-based system , “2019 11th International Conference on Knowledge and Systems Engineering (KSE) Year: 2019 | Conference Paper | Publisher: IEEE.

[2]. Nitis Monburinon , Prajak Chertchom , Thongchai Kaewkiriya , Suwat Rungpheung , Sabir Buya and Pitchayakit Boonpou, “ rediction of prices for used car by using regression models, “2018 5th International Conference on Business and Industrial Research (ICBIR) Year: 2018 | Conference Paper | Publisher: IEEE

[3]. Ning Sun , Hongxi Bai , Yuxia Geng and Huizhu Shi, “**Price evaluation model in second-hand car system based on BP neural network theory**, “2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD) Year: 2017 | Conference Paper | Publisher: IEEE