PROJECT REPORT

ON

CAR PRICE PREDICTION

Submitted by

GAURAV KUMAR

INTRODUCTION

Business Problem Framing:

In the market, we have seen lot of changes in the car prices. 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 covid19 impact, our client is facing problems with their previous car price valuation. This we have to analyse in our project.

Conceptual Background of the Domain Problem:

In this project we have to first scrape the data. In scraping we select those features what we think it is an important in our analysis and then check whether they all are important or not.

Review of Literature:

In this project we scraped the data from website (Cars24) with different locations. Then form a dataframe and save the file in csv format. In this data there are number of columns which are helpful in prediction. After that we did analysis and fit the model on selected features. Then select the best model for the car price prediction.

Motivation for the Problem Undertaken:

The seller gives there price value of the cars. And we worked on this project to predict the Car price. Whether it is okay or have variation according to the features they are given.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem:

We use Statistical techniques and analytics modeling in our projects, such as:

- describe() : use to calculate the statistical values that are mean, standard deviation, quantile deviation, minimum and maximum values. ○ corr(): use to calculate the relation between feature variable with the target variable
- Check the outliers by plotting boxplot
- skew(): use to check whether the skewness is present in the continuous data or not.

Data Source and their formats:

The data set of the Cars Price Prediction Project as show in the fig:

```
data=pd.read_csv('Used_Cars_Data.csv')
data.head()
```

	Brand	Model	Variant	Manufacturing_Year	Driven_Kilometer	Fuel	Number of Owner	Location	Price
0	Hyundai	i10	ERA 1.1 IRDE	2014	29383	Petrol	2	New Delhi	266199
1	Hyundai	i10	SPORTZ 1.2 KAPPA VTVT	2013	10558	Petrol	1	New Delhi	392199
2	Maruti	Swift	LXI	2020	10568	Petrol	1	New Delhi	592499
3	Maruti	Swift	LXI	2019	27659	Petrol	1	New Delhi	533399
4	KIA	SELTOS	HTX 1.5 PETROL MT	2020	32799	Petrol	1	New Delhi	1383099

There are 5078 rows and 9 columns. ‘Price’ column is our target variable and others are feature variables. There are 4 columns of numerical data and 5 columns of object type.

Data Pre-processing

There are no null values in the dataset but two values are object in nature so we convert it into float type. Convert object data into numerical data. There is maximum columns of object type so, we do not require to use Scaler technique.

Data inputs-Logic-Output Relationships

In correlation we see that Manufacturing_Year and Location are only columns which are positively correlated with the target variable and remaining all are negatively correlated. But there is no column having higher correlation, all are good or week.

Hardware and Software Requirements and Tools used

Hardware:

- Memory 16GB minimum
- Hard Drive SSD is preferred 500GB
- Processor intel i5 minimum
- Operating system Windows 10

Software:

- Jupyter notebook (Python)

Libraries:

pandas (used to create the data and read the data)

numpy (used with the mathematical function) seaborn (used to create a different types of graphs) matplotlib (used to plot the graph)

OrdinalEncoder (used to convert the object data into float data type) zscore (used to remove outliers)

train_test_split (split data into two parts training and testing)

r2_score (proportion of variation in independent and dependent variable)

cross_val_score (split the data into 5 folds)

mean_squared_error (how close to fit a regression line)

mean_absolute_error (calculate difference between actual and predicted value)

Model/s Development and Evaluation

Identification of possible problem:

We approach to both statistical and analytical problem

- Plot a bar graph for nominal data and distribution graph for continuous data

- ❖ describe () use to calculate mean, standard deviation, minimum, maximum and quantile deviation
- ❖ corr() used to calculate the correlation of input variable with the output variable.
- ❖ skew() used to calculate the skewness of the data
- ❖ zscore used to remove the outliers

Testing of Identified Approaches:

Here we work on the regression problem so the machine learning models are:

- Linear Regression
- Decision Tree Regression
- Random Forest Regression
- Gradient Boosting Regression

Run and Evaluate selected models:

- ➢ Linear Regression

```
lr=LinearRegression()
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.20,random_state = 42)
```

```
pred_test=lr.predict(x_test)
accuracy=r2_score(y_test,pred_test)*100
print("R square score for testing",accuracy)
```

```
R square score for testing 27.903226807274038
```

```
lrscore=cross_val_score(lr,x,y,cv=5)
lrc=lrscore.mean()
print('cross val score:',lrc*100)
```

```
cross val score: 25.725037389535288
```

```
mae=mean_absolute_error(y_test,pred_test)
mse=mean_squared_error(y_test,pred_test)
rmse=np.sqrt(mean_squared_error(y_test,pred_test))

print("Mean absolute error:",mae)
print("Mean square error:",mse)
print("Root mean square error:",rmse)
```

```
Mean absolute error: 214364.34315141593
Mean square error: 86350410040.77798
Root mean square error: 293854.40279291035
```

The r2_score is 27.90 and cv_score is 25.72. The absolute mean square is 214364.34. This score is very poor for our prediction, so this model is not appropriate for this problem.

- Decision Tree Regression

```
dtr=DecisionTreeRegressor()
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.20,random_state =38)
```

```
pred_test=dtr.predict(x_test)
dtrs=r2_score(y_test,pred_test)
print("R2 score:",dtrs*100)
```

```
R2 score: 93.96083785256997
```

```
dtrscore=cross_val_score(dtr,x,y,cv=7)
dtrc=dtrscore.mean()
print('cross val score:',dtrc*100)
```

```
cross val score: 65.67674001672786
```

```
mae=mean_absolute_error(y_test,pred_test)
mse=mean_squared_error(y_test,pred_test)
rmse=np.sqrt(mean_squared_error(y_test,pred_test))

print("Mean absolute error:",mae)
print("Mean square error:",mse)
print("Root mean square error:",rmse)
```

```
Mean absolute error: 20801.574803149608
Mean square error: 8210956587.746063
Root mean square error: 90614.32882136281
```

The r2_score is 93.96 and cv_score is 65.67. The mean absolute error is 20801.57. In Decision Tree Regression there is good r2_score.

- Random Forest Regression

```
rfr=RandomForestRegressor()
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.20,random_state =12)
```

```
preds_test=rfr.predict(x_test)
rfrs=r2_score(y_test,preds_test)
print("R2 score for testing:",rfrs*100)
```

```
R2 score for testing: 96.28400745545888
```

```
rfrscore=cross_val_score(rfr,x,y,cv=9)
rfrc=rfrscore.mean()
print('cross val score:',rfrc*100)
```

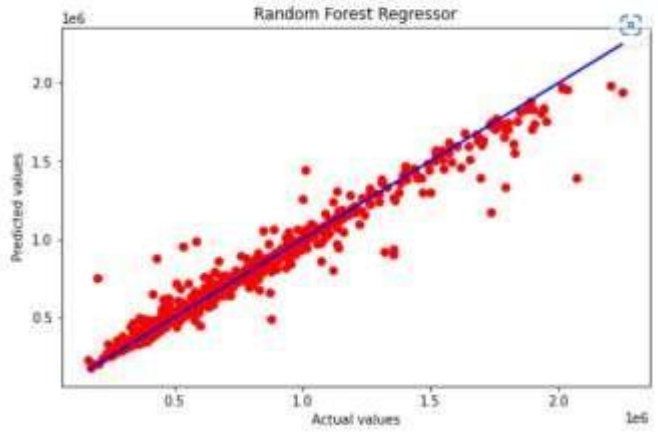
```
cross val score: 81.20064648139179
```

```
mae=mean_absolute_error(y_test,preds_test)
mse=mean_squared_error(y_test,preds_test)
rmse=np.sqrt(mean_squared_error(y_test,preds_test))

print("Mean absolute error:",mae)
print("Mean square error:",mse)
print("Root mean square error:",rmse)
```

```
Mean absolute error: 33256.82393700788
Mean square error: 5105908706.830263
Root mean square error: 71455.64153256385
```

```
plt.figure(figsize=(8,5))
plt.scatter(y_test,preds_test,color='red')
plt.plot(y_test,y_test,color='blue')
plt.xlabel('Actual values')
plt.ylabel('Predicted values')
plt.title('Random Forest Regressor')
plt.show()
```



The r2_score is 96.28 and cv_score is 81.20. The mean absolute score is 33256.82. On plotting the graph we see that there is less difference between actual and predicted value.

- Gradient Boosting Regression

```
gb= GradientBoostingRegressor()
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.20,random_state = 5)
```

```
pred_test=gb.predict(x_test)
accuracy=r2_score(y_test,pred_test)*100
print("R square score",accuracy)
```

```
R square score 82.93977308028634
```

```
gbscore=cross_val_score(gb,x,y,cv=8)
gbc=gbscore.mean()
print('cross val score:',gbc*100)
```

```
cross val score: 75.20991324700115
```

```
mae=mean_absolute_error(y_test,pred_test)
mse=mean_squared_error(y_test,pred_test)
rmse=np.sqrt(mean_squared_error(y_test,pred_test))

print("Mean absolute error:",mae)
print("Mean square error:",mse)
print("Root mean square error:",rmse)
```

```
Mean absolute error: 98242.59609002623
Mean square error: 21093536531.11875
Root mean square error: 145236.14058187703
```

The r2_score is 82.93 and cv_score is 75.20. The mean absolute score is 98242.59. This model is good but not that much.

Visualisation:

On visualising the data we see that there are large demand of Maruti's car which is about 40%, Hyundai is about 20% and Honda is about 10% and left of all the eighteen brands are under the remaining 30%. Mostly customers prefer Petrol car as compare to Diesel and CNG. We see that persons of New Delhi, Gurgaon, Noida and Bengaluru are fastly change their cars. There are 80% cars resale by first Owner. Most of the cars driven upto 100000 Km and very few driven more than this.

Interpretation of the Results:

On visualising we see that most of the person prefer Maruti cars in petrol. They mostly drive upto 100000 km and then they resale it. In the data there is most of the features are categorical in nature. After fitting the models we see that Random Forest Regression is the better on who give better score and have minimum error. There is less difference between actual and predicted values.

CONCLUSION

Key Findings and Conclusions of the Study:

On study the problem we get to know that on reselling the cars all the features are effected on our analysis whether it is less or more. 75% cars are sell which are manufactured in last 6 years.

Learning Outcomes of the Study in respect of Data Science:

On analysing the data we study that Maruti car is mostly preferable by the person in petrol. Cars are mostly resell by the first owner. In this problem we see that there is less correlation between independent and dependent variable. In column Driven_Kilometer has skewness so, we remove them by transformation. After that we fit the model and select the Random Forest Regression is the better model for Car Price Prediction.

Limitations of this work and Scope for Future Work:

Limitations:

- The chances of getting no warranty, if get it will be very limited.
- On used cars the maintenance cost is high comparatively new cars.

Scope:

- By this anyone can get to know that what the price of cars they are searching for.