Project: Used Cars Price Prediction

By Aishwary Shukla

ACKNOWLEDGMENT

The aim of this project was to predict used Cars prices using Machine Learning. A data set was provided. From this dataset models were built in Jupyter notebook using Python and ML libraries. After a preliminary study of the available algorithms and data review, it became apparent that the problem fell under regression category. Thus, we have used regression algorithms; linear regression, decision tree, Random Forest, KNeighbours Regressor, Support Vector Machine, AdaBoostRegressor respectively. The major discovery is that the machine learning approach should be suitable for these types of problems due to many aspects. Python programming language and its libraries namely Pandas, Numpy, Matplotlib, Seaborn etc are also a good choice for a first step, not the least because of the easily grasped user interface, as well as the wide availability of algorithms within machine learning. Advanced methods such as hyper parameter tuning of best models is also available.

Data References:

- https://cars24.com/

Special thanks to:

Mr. Shankar Gaud Tegimanni, DataTrained

Mr. Shwetank Mishra, FlipRobo

Mr. Prateeek Rajvanshi, Clevered Institute

And all colleagues, mentors, trainers from DataTrained and FlipRobo for training me and giving me the opportunity to be an intern and expand my knowledge in the field of Data Science and Artificial Intelligence.

INTRODUCTION

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.

Abstract

Due to the unprecedented number of cars being purchased and sold, used car price prediction is a topic of high interest. Because of the affordability of

used cars in developing countries, people tend more purchase used cars. A primary objective of this project is to estimate used car prices by using attributes that are highly correlated with a label (Price). To accomplish this, data mining technology has been employed. Null, redundant, and missing values were removed from the dataset during pre-processing. In this supervised learning study, three regressors (Random Forest Regressor, Linear Regression, and XGBoostRegressor) have been trained, tested, and compared against a benchmark dataset.

Among all the experiments, the Random Forest Regressor had the highest score at 95%, followed by 0.025 MSE, 0.0008 MAE, and 0.0378 RMSE respectively. In addition to Random Forest Regression, Bagging Regression performed well with an 88% score, followed by Linear Regression having an 85% mark. A train-test split of 80/20 with 40 random states was used in all experiments. The researchers of this project anticipate that in the near future, the most sophisticated algorithm is used for making predictions, and then the model will be integrated into a mobile app or web page for the general public to use.

This project contains two phases:

1. Data Collection Phase
2. Model Building Phase

History of Used Cars price prediction and its Importance:

Several studies and related works have been done previously to predict used car prices around the world using different methodologies and approaches, with varying results of accuracy from 50% to 90%. In (Pudaruth, 2014) the researcher proposed to predict used car prices in Mauritius, where he applied different machine learning techniques to achieve his results like decision tree, K-nearest neighbours, Multiple Regression and Naïve Bayes algorithms to predict the used cars prices, based on historical data gathered from the newspaper. Achieved results ranged from accuracy of 60-70

percent, the author suggested using more sophisticated models and algorithms to make the evaluation, with the main weakness off the decision tree and naïve Bayes that it is required to discretize the price and classify it which accrue to more inaccuracies. Moreover, he suggested a larger set of data of data to train the models hence the data gathered was not sufficient. (Monburinon, et al., 2018) Gathered data from a German e-commerce site that totalled to 304,133 rows and 11 attributes to predict the prices of used car using different techniques and measured their results using Mean Absolute Error (MEA) to compare their results. Same training dataset and testing dataset was given to each model. Highest results achieved was by using gradient boosted regression tree with a MAE of 0.28, and MEA of 0.35 and 0.55 for mean absolute error and multiple linear regression respectively. Authors suggested adjusting the parameters in future works to yield better results, as well as using one hot encoding instead of label encoding for more realistic data interpretations on categorical data. (Gegic, Isakovic, Keco, Masetic, & Kevric, 2019) from the International Burch University in Sarajevo, used three different machine learning techniques to predict used car prices. Using data scrapped from a local Bosnian website for used cars totalled at 797 car samples after pre-processing, and proposed using these methods: Support Vector Machine, Random Forest and Artificial Neural network. Results have shown using only one machine learning algorithm achieved results less than 50%, whereas after combing the algorithms with pre calcification of prices using Random Forest, results with accuracies up to 87.38% was recorded. (Noor & Jan, 2017) were able to achieve high level of accuracy using Multiple linear regression models to predict the price of cars collected from used cars website in Pakistan called Pak Wheels that totalled to 1699 records after pre-processing, and where able to achieve accuracy of 98%, this was done after reducing the total amount of attributes using variable selection technique to include significant attributes only and to reduce the complexity of the model. 6 (K.Samruddhi & Kumar, 2020) Proposed using Supervised machine leaning model using K-Nearest Neighbour to predict used car prices from a data set obtained from Kaggle containing 14 different attributes, using this method accuracy reached up to 85% after different values of K as well as Changing the percent of training data to testing data, expectedly when increasing the percent of data that is tested better accuracy results are achieved. The model was also cross validated with 5 and 10 folds by using K fold method. (Gongqi, Yansong, &

Qiang, 2011) proposed using Artificial Neural Network (ANN) through a combined method of BP neural network and nonlinear curve fit and have achieved accurate value prediction with a feasible model. (Listiani, 2009) used Support Vector Machines to evaluate leased cars prices, results have shown that SVM is far more accurate in large dataset with high dimensional data than Multiple linear regression. Whereas the computation Multiple linear regression can take several minutes and the SVM would take up to a day to compute the results. Multiple linear regression may be simple, but SVM is far more accurate. Moreover, the study includes Samples with up to 178 attributes which is far more than the proposed variable in our study, hence the use of multiple linear regression may be more suitable in our case. (Kuiper, 2008) Collected data from General Motor of cars that are produced in 2005, where he as well used variable selection technique to include the most relevant attributes in his model to reduce the complexity of the data. He proposed used Multivariate regression model that would be more suitable for values with numeric format. In order to predict the price of used cars, researchers (Nabarun Pal, 2018) used a supervised learning method known as Random Forest. Kaggle's dataset was used as a basis for predicting used car prices. In order to determine the price impact of each feature, careful exploratory data analysis was performed. 500 Decision Trees were trained with Random Forests.

It is most commonly used for classification, but they turned it into a regression model by transforming the problem into an equivalent regression problem. Using experimental results, it was found that training accuracy was 95.82%, and testing accuracy was 83.63%. By selecting the most correlated features, the model can accurately predict the car price. In light of the number of works that have been done in this field, another group of researchers (Jian Da Wu, 2017) conducted research on this topic and tried to develop a system that consists of three components: a data acquisition system, a price forecasting algorithm, and a performance analysis. Due to its adaptive learning capability, a conventional artificial neural network (ANN) with a back-propagation network is compared to the proposed ANFIS. In the ANFIS, qualitative fuzzy logic approximation as well as adaptive neural network capabilities are included. Using ANFIS as an expert system in predicting used car prices showed better results in the experiment. Using

GUI, the consumer can get accurate and convenient 7 information about used cars' purchasing prices, and experiments proved that the proposed system could provide accurate and convenient price forecasting. Hence, from all literature review it is concluded that used cars price prediction is an important topic which is the area of many researchers nowadays. So far, the best achieved accuracy is 83.63% on kaggle's dataset using random forest technique. The researchers have tested multiple regressors and final model is regression model using linear regression.

Conceptual Background of the Domain Problem

Background Today, the transportation industry is considered to be one of the backbones of the economy. Automobiles are referred to as the "Industry of Industries" in developed nations. According to industry professionals, the India's automotive industry has seen remarkable growth. Besides being the fastest-growing nation in the automobile industry, it represents its global presence. In Dubai, like most other countries, cars are gaining a great deal of popularity among the local population and the ex-pat community who work in the country. There are used cars for sale in the India of all makes and models, even cars from well-known brands (Maruti, 2019). India's auto industry is experiencing constant growth. So far, the market in the India has grown by 19%. It is thus the world's largest market in terms of growth rate. Almost everyone wants their own car these days, but because of factors like affordability or economic conditions, many prefer to opt for pre-owned cars. Accurately predicting used car prices requires expert knowledge due to the nature of their dependence on a variety of factors and features. Used car prices are not constant in the market, both buyers and sellers need an intelligent system that will allow them to predict the correct price efficiently. In this intelligent system, the most difficult problem is the collection of the dataset which contains all important elements like the manufacturing year of the car, its gas type, its condition, miles driven, horsepower, doors, number of times a car has been painted, customer reviews, the weight of the car, etc. It is clear that the price of the product is affected by many factors, but unfortunately, information about these features is not always readily available. Since this project primarily focuses on the India market, the

benchmark dataset containing most key features is scraped. It is necessary to pre-process and transform collected data in the proper format prior to feeding it directly to the data mining model. As a first step, the dataset was statistically analysed and plotted.

Missing, duplicated, and null values were identified and dealt with. Features were chosen and extracted using 2 correlation matrices. To build an efficient model, the most correlated features were retained, and others were discarded. This prediction problem can be considered a regression problem since it belongs to the supervised learning domain. Three Regressor known as random forest, linear regression, and bagging regression were trained and compared. A random forest Regressor outperformed all others in this project, so it was chosen as the main algorithm model.

Review of Literature

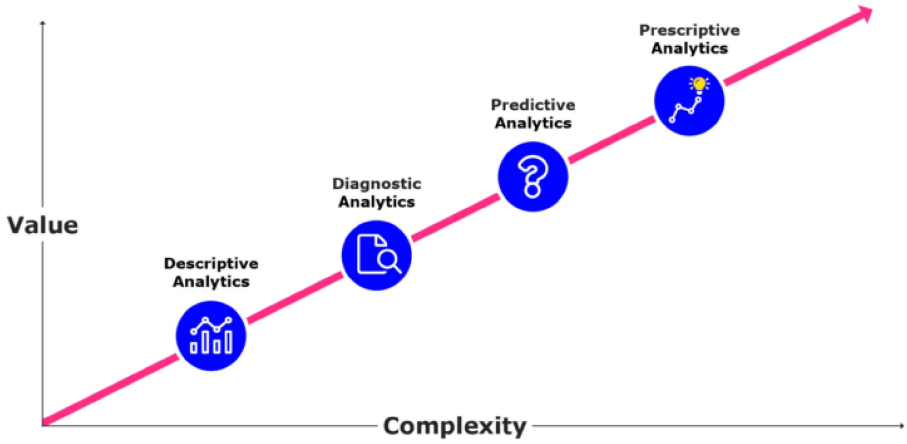
Steps followed to conduct the research of finding important features that positively affect the fares of flights.

- Exploratory data analysis
- Feature Engineering
- Relation between different features/amenities of house and target column i.e., sales price.
- Grouped Data Analysis to get more insights.
- Correlation and identification of multicollinearity problem.
- Identifying and selecting best features that affect fare price.
- Data pre-processing.
- Model Initialization and evaluation.
- Model Validation
- Model Testing and Pipelining.

Motivation for the Problem Undertaken

My objective behind making this project is to implement techniques and methods I have learned and practiced during my PG programme in Data Science. Further I continuously strive to improve my skills, adapt new techniques and methods in data analysis and modelling. Travel and tourism interest me thus working on this project will give me domain knowledge and technical expertise in deploying ml models for solving real world problems.

Analytical Problem Framing



Analytical Modeling and Machine Learning are typically regarded as two alternative methodologies to model performance of computer systems and applications. In this paper we have overviewed three different hybrid modeling methodologies, which leverage on both AM and ML to get the best of the two worlds, namely reducing the model's training time and increasing its accuracy as new training data become available. While research on AM and ML has already reached maturity, investigation on hybrid methodologies is still at its infancy.

A recent work has shown that none of such techniques outperforms the others in terms of accuracy for every application and for every training data set. An interesting research line to pursue, in the light of this result, is to identify which characteristics of the applications being modeled, or of the AM and ML techniques employed for modeling may lead a given hybrid methodology to outperform the others. Another interesting issue to investigate is whether it is possible to fruitfully combine further the described methodologies, with the goal of building a unique, more accurate, meta-hybrid model.

Data Sources and their formats

Data was collected and Scrapped from a website that sell car in the India called Cars24, by using a scrapping library available selenium, through multiple runs and iterations 5000+ rows of data with 23 variables were collected successfully from the website. At first, the data types of each attribute were corrected/converted by performing pre-processing on each attribute individually. Details and description of dataset is described in table below:

Downpayment	City	Brand Name	Model Name	Year	Type	Kms Driven	Plate Number State	Plate Number	Full amount	...	Owner Details	Fuel Type	last_service kms	last_service day	last_service month
Zero downpayment	New Delhi	Maruti	Swift LXI	2020	MANUAL	22826.0	HR	52	551000.0	...	1st	Petrol	22826	10	Jan
Zero downpayment	New Delhi	KIA	SELTOS GTX+ 1.4 MT	2020	MANUAL	5999.0	HR	26	0.0	...	1st	Petrol	5999	4	Jan
Zero downpayment	New Delhi	Maruti	Swift LXI	2020	MANUAL	16691.0	HR	51	557000.0	...	2nd	Petrol	16691	26	Oct
Zero downpayment	New Delhi	KIA	SELTOS GTX + AT PETROL	2020	AUTOMATIC	9417.0	DL	8C	1864000.0	...	1st	Petrol	9417	13	Jan
Zero downpayment	New Delhi	Hyundai	Grand i10 MAGNA 1.2 KAPPA VTVT	2019	MANUAL	19964.0	DL	10	530000.0	...	1st	Petrol	19964	18	Dec
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Zero downpayment	Gurgaon	Toyota	Corolla Altis VL AT	2011	AUTOMATIC	24133.0	UP	14	0.0	...	1st	Petrol	24133	1	Feb
Zero downpayment	Gurgaon	Maruti	Swift Dzire VXI	2011	MANUAL	82915.0	DL	9C	0.0	...	1st	Petrol	82915	31	Jan
Zero downpayment	Gurgaon	Renault	Kwid 1.0 RXT Opt	2018	MANUAL	41823.0	DL	8C	0.0	...	1st	Petrol	41823	31	Jan

```
: Downpayment                 object
  City                        object
  Brand Name                  object
  Model Name                  object
  Year                         int64
  Type                        object
  Kms Driven                 float64
  Plate Number State          object
  Plate Number                object
  Full amount                float64
  Discounted amount          float64
  Monthly EMI                float64
  History                     object
  Owner Details               object
  Fuel Type                   object
  last_service kms             int64
  last_service day             int64
  last_service month          object
  last_service year            int64
  Insurance validity Month    object
  Insurance validity Year      int64
  Registration Month          object
  Registration Year            int64
  dtype: object
```

```
Index(['Year', 'Kms Driven', 'Full amount', 'Discounted amount', 'Monthly EMI',
       'last_service kms', 'last_service day', 'last_service year',
       'Insurance validity Year', 'Registration Year'],
      dtype='object')
Index(['Downpayment', 'City', 'Brand Name', 'Model Name', 'Type',
       'Plate Number State', 'Plate Number', 'History', 'Owner Details',
       'Fuel Type', 'last_service month', 'Insurance validity Month',
       'Registration Month'],
      dtype='object')
```

Data Pre-processing Done

Raw data was converted into a comprehensible format. There is often a lack of specific activity or trend data, and many inaccurate facts are included in real-world data. Consequently, this may result in poor-quality data collection, and, in turn, poor-quality models constructed from the data. Such problems can be resolved by pre-processing the data. Pre-processing in Machine Learning is the process of modifying, or encoding, data so that the machine can parse it more easily. Thus, the algorithm can now properly interpret the data.

Steps included in data pre-processing are mentioned below:

1. Reorganising And cleaning of data

The data that we have scrapped is mixed up and it needs to be reorganised and distributed into rows so that each and every feature can be distinguished and understood clearly.

```
df.drop(columns = ['last_service'],axis=1,inplace=True)
```

```
df['Registration Year'].replace('NA',2019, inplace=True)
```

```
df['last_service day'] = df['last_service day'].astype('int64')
df['last_service year'] = df['last_service year'].astype('int64')
df['Insurance validity Year'] = df['Insurance validity Year'].astype('int64')
df['Registration Year'] = df['Registration Year'].astype('int64')
```

```
clfa = []
for i in df['Full amount']:
    if i != '0':
        clfa.append(i.replace(',', '').replace('₹', ''))
    elif i == 'Fixed Price':
        clfa.append('Fixed Price')

df['Full amount'] = clfa
```

```
# Treating full price:

df['Full amount'].replace('Fixed Price', 0, inplace=True)
```

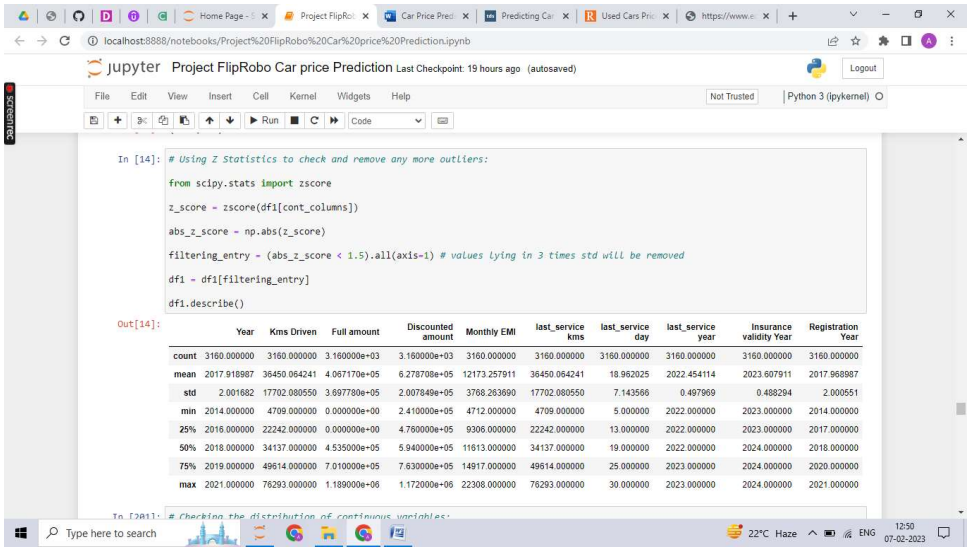
```
df['Full amount'] = df['Full amount'].astype('float64')
```

Moreover, data also contains some missing values we have already cleaned while organising the data and further we have 5000 rows of data that we can further use for our data analysis.

2. Identification and removal of outliers:

We have used Z-Score method which is essentially how many standard deviations away is my actual value from the mean value based on the business context,

we can define the **threshold** value for the z score to classify a point as an outlier or not in the current scheme of things. Computation and removal can be seen below:



abs() is one of the simplest pandas data frame function. It returns an object with the absolute value taken and it is only applicable to objects that are all numeric. It does not work with any Nan value either. abs() function can also be used with complex numbers to find their absolute value.

3. Data normalization:

Data normalization is a technique used in data mining to transform the values of a dataset into a common scale. This is important because many machine learning algorithms are sensitive to the scale of the input features and can produce better results when the data is normalized.

Here we have used Power transforms are **a family of parametric, monotonic transformations that are applied to make data more Gaussian-like**. This is useful for modelling issues related to heteroscedasticity (non-constant variance), or other situations where normality is desired.

4. Multicollinearity analysis using Variance Inflation Factor:

Multicollinearity occurs when two or more independent variables are highly correlated with one another in a regression model.

```
from statsmodels.stats.outliers_influence import variance_inflation_factor

vif = pd.DataFrame()
vif['vif'] = [variance_inflation_factor(df1[cont_columns],i) for i in range(df1[cont_columns].shape[1])]
vif['features'] = df1[cont_columns].columns
vif.sort_values(by='vif',ascending=False)
```

	vif	features
1	inf	Kms Driven
5	inf	last_service kms
9	8.343717e+07	Registration Year
0	8.309426e+07	Year
8	8.735824e+06	Insurance validity Year
7	8.413976e+06	last_service year
4	1.197638e+04	Monthly EMI
3	1.124627e+04	Discounted amount
6	8.162490e+00	last_service day
2	2.768624e+00	Full amount

Here we have computed vif to pin point variables contributing to multicollinearity.

” VIF determines the strength of the correlation between the independent variables. It is predicted by taking a variable and regressing it against every other variable. “

or

VIF score of an independent variable represents how well the variable is explained by other independent variables.

5. Data reduction

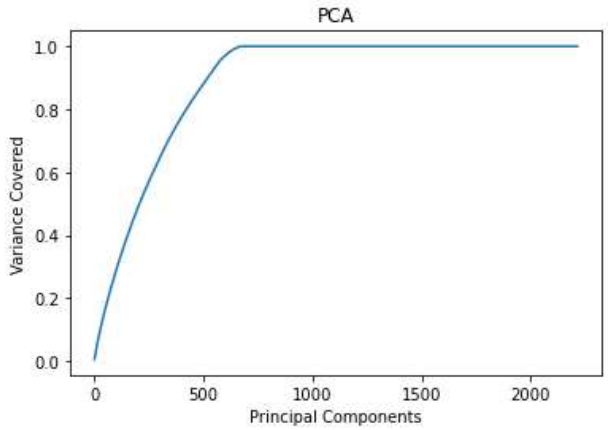
Dimensionality reduction, or dimension reduction, is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful properties of the original data.

Principal Component Analysis (PCA) is a statistical procedure that uses an orthogonal transformation that converts a set of correlated variables to a set of uncorrelated variables. PCA is the most widely used tool in exploratory data analysis and in machine learning for predictive models. Moreover, PCA is an unsupervised statistical technique used to examine the interrelations among a set of variables. It is also known as a general factor analysis where regression determines a line of best fit.

Here we have performed data reduction using principal component analysis by plotting a scree plot which shows an estimate of the number of components that we should consider for training our model.

```
: # Using Scree Plot to identify best components:

plt.figure()
plt.plot(np.cumsum(pcaf.explained_variance_ratio_))
plt.xlabel('Principal Components')
plt.ylabel('Variance Covered')
plt.title('PCA')
plt.show()
```



Data Inputs- Logic- Output Relationships

Describe the relationship behind the data input, its format, the logic in between and the output. Describe how the input affects the output.

Hardware and Software Requirements and Tools Used

Hardware for Machine Learning

- Computer/Intel's CPU/Powerful GPU enabled CPU

Hardware for Machine Learning

- Jupyter Notebook/Anaconda
- Scikit learn
- Pandas
- Numpy
- Matplotlib
- Seaborn

Model/s Development and Evaluation

Approaches followed, both statistical and analytical, for solving of this problem

Algorithms used

1. Linear regression
2. Ridge and lasso regression
3. decision tree regression
4. random forest regressor
5. xg boost regressor
6. ada boost regressor
7. support vector regressor
8. K neighbours regressor

Models Build (Best)

Random Forest Regressor

```
# Random forest:

from sklearn.ensemble import RandomForestRegressor
regressor_rf = RandomForestRegressor(n_estimators=1200,
                           max_depth=15,
                           min_samples_split=5,
                           min_samples_leaf=5,
                           max_features=None,
                           oob_score=True,
                           random_state=42)

regressor_rf.fit(x_train, y_train)

lr_normal_rf = regressor_rf.score(x_train, y_train)

lr_normal_rf
```

```
0.945142554766031
```

```
y_predrf = regressor_rf.predict(x_test)

lr_normal_rf_test = regressor_rf.score(x_test, y_test)

lr_normal_rf_test

mse_lr_normal_rf  = mean_absolute_error(y_test, y_predrf)

mse_lr_normal_rf
```

```
76450.17890370278
```

XG Boost Regressor

```
# Hyperparameter tuning in XGBoost:

xgb_reg1 = xgb.XGBRegressor(learning_rate=0.05,
                        n_estimators=6000,
                        max_depth=4,
                        min_child_weight=0,
                        gamma=0.6,
                        subsample=0.7,
                        colsample_bytree=0.7,
                        objective='reg:linear',
                        nthread=-1,
                        scale_pos_weight=1,
                        seed=27,
                        reg_alpha=0.00006,
                        random_state=42)
xgb_reg1.fit(x_train,y_train)
y_predx = xgb_clf.predict(x_test)

# Model Evaluation: MAE , MSE , RMSE

from sklearn.metrics import mean_absolute_error,mean_squared_error
print('tr accu',xgb_reg1.score(x_train, y_train))
print("MAE: ", metrics.mean_absolute_error(y_test, y_predx))
print("MSE: ", metrics.mean_squared_error(y_test, y_predx))
print("RMSE: ", metrics.mean_squared_error(y_test, y_predx, squared=False))
print("R2: ", metrics.r2_score(y_test, y_predx), "\n")
print("Score: ", xgb_reg1.score(x_test, y_predx))
```

```
[16:03:06] WARNING: C:/Users/administrator/workspace/xgboost-win64_release_1.6.0/src/objective/regression_obj.cu:203: reg:linea
r is now deprecated in favor of reg:squarederror.
tr accu 0.9999999997798875
MAE:  50233.97339794304
MSE:  5966814467.414483
RMSE:  77245.15821340831
R2:  0.8531789654874624
```

Key Metrics/ Parameters for success in solving problem under consideration

Here we have used following parameters for evaluating our regression models.

1. Mean Square Error (MSE): MSE is the single value that provides information about goodness of regression line. Smaller the MSE value, better the fit because smaller value implies smaller magnitude of errors.

$$\boldsymbol{MSE = 1\ N \sum |yi - y|\ 2\ N\ i=1}$$

2. Root Mean Square Error (RMSE): RMSE is the quadratic scoring rule that also measures the average magnitude of the error. It is the square root of average squared difference between prediction and actual observation.

3. Mean Absolute Error (MAE): This measure represents the average absolute difference between the actual and predicted values in the dataset. It represents the average residual from the dataset.

$$MAE = 1\ N\ \sum|yi - y|$$

4. R-Square/Adjusted R-Square:

TSS gives us the total variation in Y, and RSS gives us the variation in Y not explained by X, then **TSS-RSS gives us the variation in Y that is explained by our model**! We can simply divide this value by TSS to get the proportion of variation in Y that is explained by the model. And this our **R-squared statistic**

$$\text{R-squared} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

The Adjusted R-squared takes into account the number of independent variables used for predicting the target variable. In doing so, we can determine whether adding new variables to the model actually increases the model fit.

Let's have a look at the formula for adjusted R-squared to better understand its working.

Here,

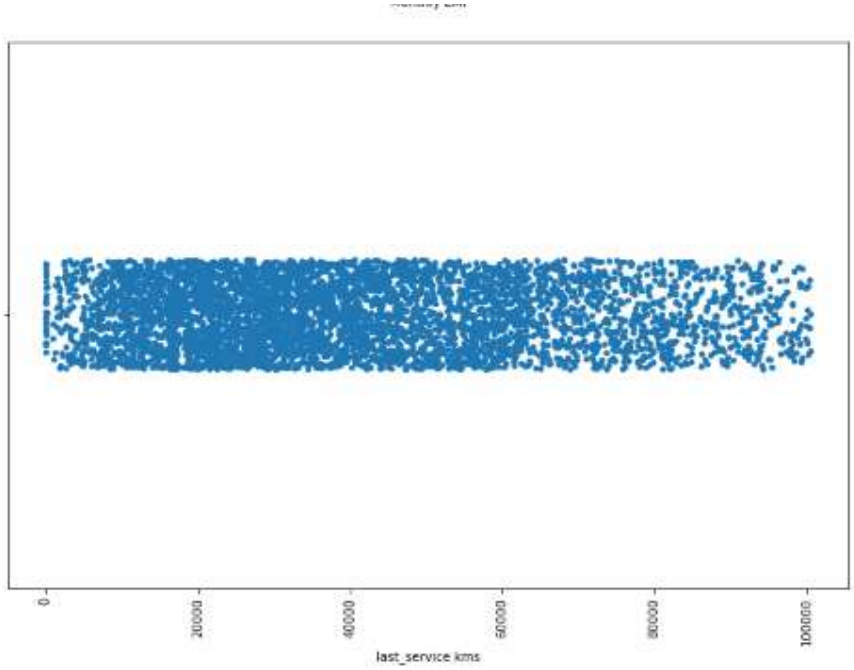
- **n** represents the number of data points in our dataset
- **k** represents the number of independent variables, and
- **R** represents the R-squared values determined by the model.

By evaluation of our models using above mentioned matrix parameters it is clearly visible that linear regression and other regularization techniques and decision tree etc not performed well or it can also be said as they have failed to understand the variance in the data set.

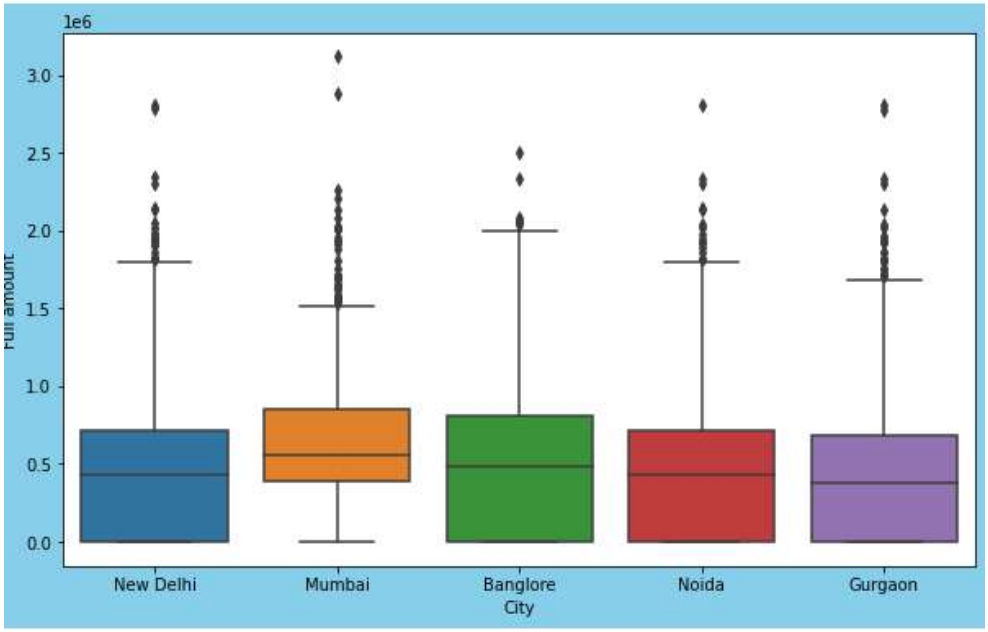
This can be due to the reason that the data set has wide range of data and contains a lot of outliers. by using the ensemble techniques of the random forest regressor model and the xg boost model that is a gradient boosting technique have performed very well in predicting the prices of used cars in comparison to other models.

Visualizations

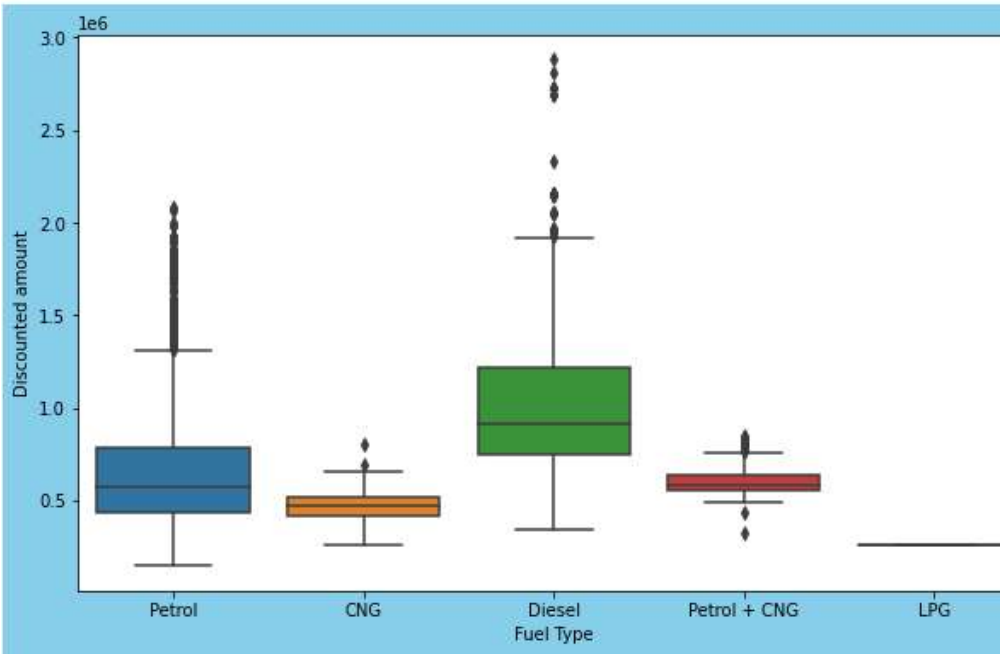
Strip plot



Box plot

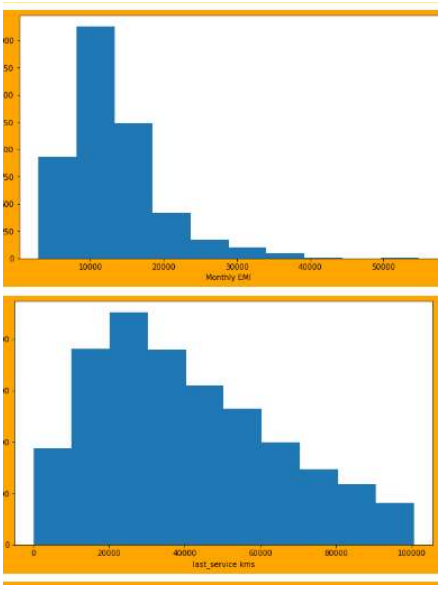


Full amount of car prices in metropolitan cities like New Delhi Mumbai Gurgaon Noida are more than average and contain lot of outliers.

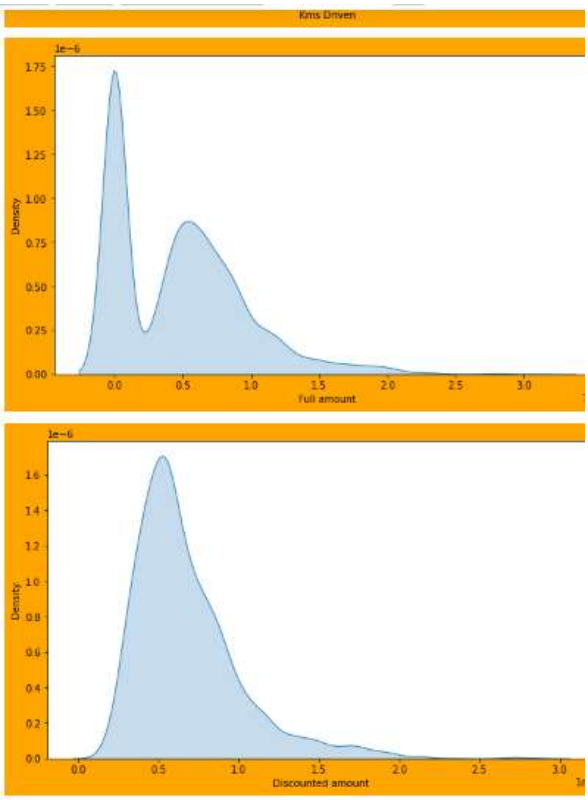


By box plots, Skoda and Fiat brand cars have relatively less outliers, registration year feature also has very less outliers, petrol and diesel cars have the most outliers when plotted against the label column that is amount.

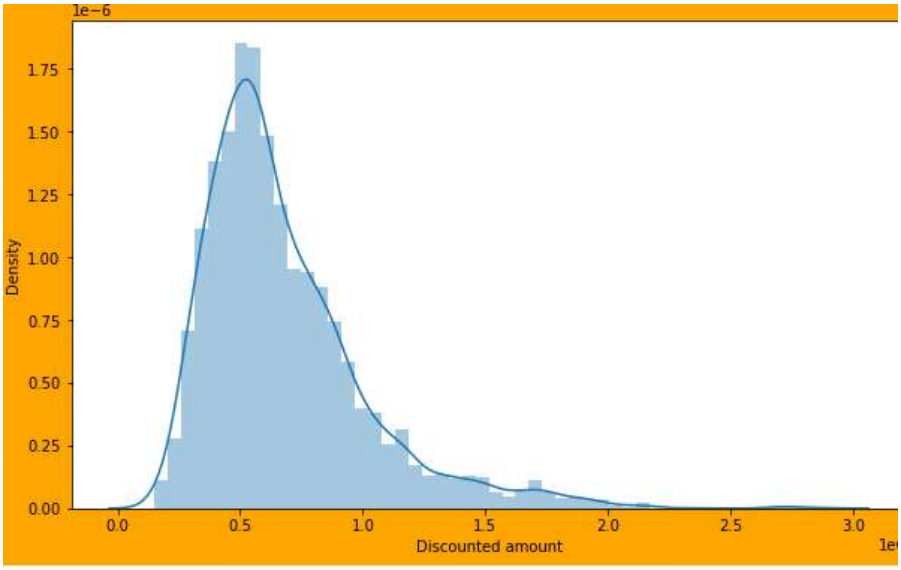
Histogram



Kernel destiny estimation plot

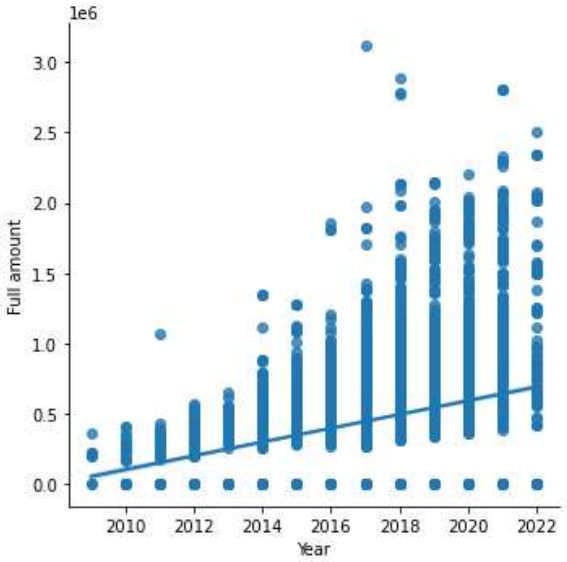


Distribution plot



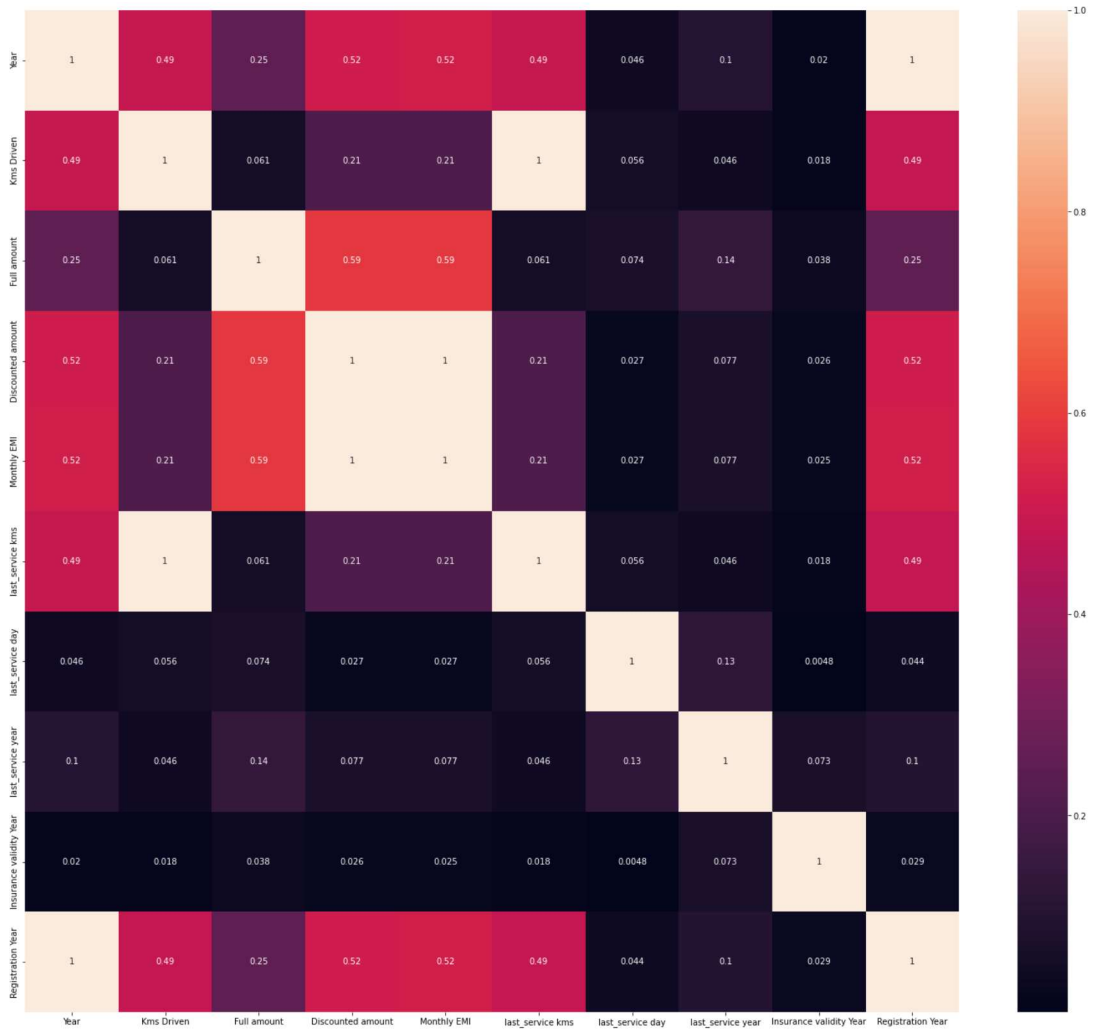
Distribution plot depicts that features except full amount and discounted amount of used cars which are varying a lot and rightly skewed and contain a lot of outliers, rest features are Slightly skewed and within permissible range of skewness.

Lmplot



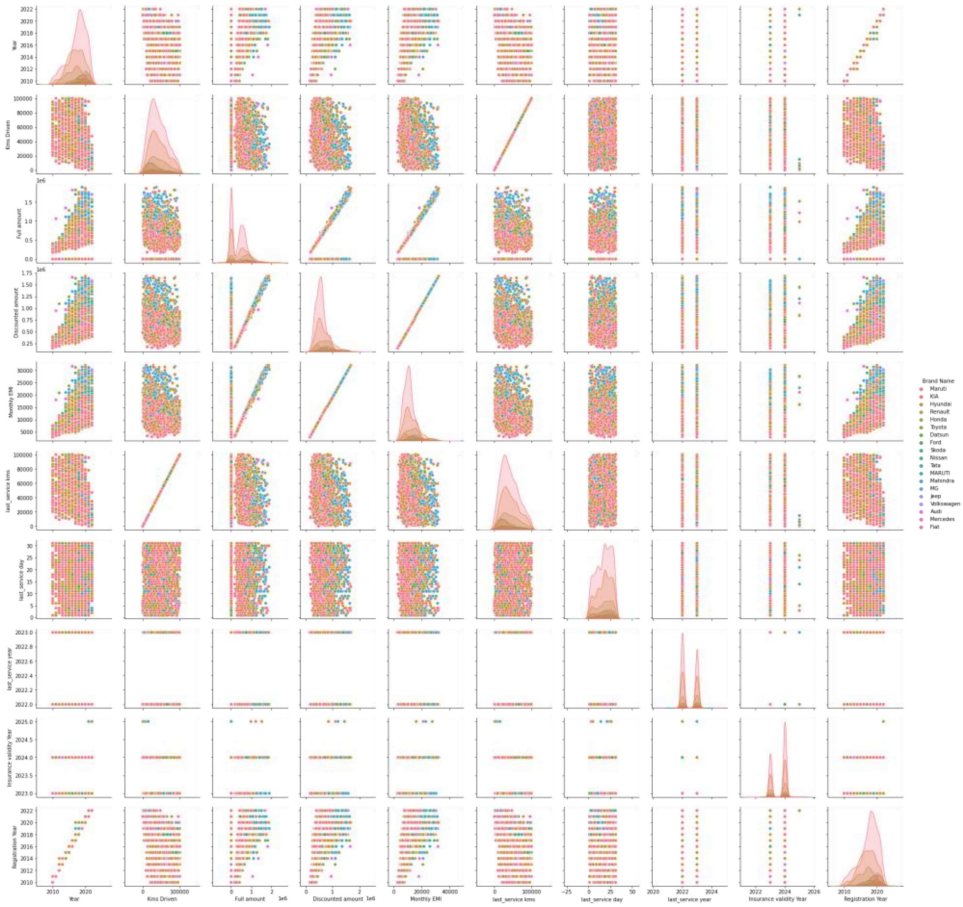
By performing the regression analysis between the features and the label column here we can find that year feature, Monthly EMI full amount, registration year show moderate to high positive relation with target column that is discounted amount.

Heat map



Heat map helps us to analyse the collinearity between the features and with the help of above plotted heat map we can identify that registration year service columns, price columns, kilometres driven column Are highly correlated with the other features but OK Goog

Pairplot



Mg Audi Maruti Jeep and kia are the most expensive in used cars collection. Ford Skoda mg used cars for sale are very less as compared to Indian brands like Maruti. prices of SUVs are higher.

Grouped Data Analysis

1. Grouping by brands:

```
[75]: brand_groups['Full amount'].mean().sort_values(ascending=False)
```

```
t[75]: Brand Name
MG              1.638944e+06
Audi            1.533600e+06
MARUTI          1.469000e+06
Jeep            1.362467e+06
KIA             1.149475e+06
Mercedes        1.064000e+06
Skoda           1.001587e+06
Mahindra        9.825036e+05
Toyota          7.987443e+05
Tata            7.007510e+05
Volkswagen      6.926479e+05
Ford            5.968031e+05
Honda           5.688822e+05
Nissan          4.663077e+05
Renault         4.270908e+05
Hyundai         3.984598e+05
Maruti          3.440808e+05
Datsun          2.303913e+05
SKODA           0.000000e+00
Fiat            0.000000e+00
Name: Full amount, dtype: float64
```

```
[190]: brand_groups['Full amount'].median().sort_values(ascending=False)
```

```
[190]: Brand Name
MG              1714000.0
MARUTI          1497000.0
Jeep            1388000.0
KIA             1378000.0
Audi            1346000.0
Mahindra        1169000.0
Mercedes        1064000.0
Skoda           1037000.0
Toyota           881000.0
Ford             705000.0
Tata             676000.0
Honda            644000.0
Volkswagen       612000.0
Nissan           561000.0
Renault          399000.0
Hyundai          390000.0
Maruti           387000.0
Datsun           302000.0
Fiat                  0.0
SKODA                 0.0
Name: Full amount, dtype: float64
```

```
brand_groups['Year'].mean().sort_values(ascending=False)
```

```
Brand Name
SKODA         2022.000000
MARUTI        2022.000000
KIA           2019.979798
Tata          2019.715953
MG            2019.611111
Mahindra      2019.248175
Renault       2019.003984
Skoda         2018.913043
Datsun        2018.608696
Nissan        2018.384615
Jeep          2018.000000
Maruti        2017.467933
Volkswagen    2017.408451
Toyota        2017.375940
Ford          2017.171875
Hyundai       2016.969522
Honda         2016.656371
Fiat          2016.500000
Audi          2014.800000
Mercedes      2011.000000
Name: Year, dtype: float64
```

```
brand_groups['Year'].median().sort_values(ascending=False)
```

```
Brand Name
SKODA         2022.0
MARUTI        2022.0
Tata          2020.0
Skoda         2020.0
KIA           2020.0
Mahindra      2019.0
MG            2019.0
Renault       2019.0
Nissan        2018.0
Toyota        2018.0
Jeep          2018.0
Datsun        2018.0
Maruti        2018.0
Volkswagen    2017.0
Hyundai       2017.0
Honda         2017.0
Ford          2017.0
Fiat          2016.5
Audi          2014.0
Mercedes      2011.0
Name: Year, dtype: float64
```

```
brand_groups['Year'].count().sort_values(ascending=False)
```

```
Brand Name
Maruti        2105
Hyundai       1214
Honda          518
Tata           257
Renault        251
Mahindra       137
Toyota         133
Ford           128
KIA             99
Volkswagen      71
Skoda           46
Datsun          23
MG              18
Jeep            15
Nissan          13
Audi             5
MARUTI           4
Fiat             2
Mercedes         1
SKODA            1
```

2. Grouping by Fuel type:

```
fuel_groups = df.groupby('Fuel Type')
```

```
fuel_groups['Full amount'].mean().sort_values(ascending=False)
```

```
Fuel Type
Diesel          810138.595029
Petrol          443242.997147
CNG             288068.000000
Petrol + CNG    227520.000000
LPG                  0.000000
Name: Full amount, dtype: float64
```

```
fuel_groups['Full amount'].median().sort_values(ascending=False)
```

```
Fuel Type
Diesel          843000.0
Petrol          439000.0
CNG             417000.0
LPG                  0.0
Petrol + CNG         0.0
Name: Full amount, dtype: float64
```

```
fuel_groups['Discounted amount'].mean().sort_values(ascending=False)
```

```
Fuel Type
Diesel          1.025756e+06
Petrol          6.381867e+05
Petrol + CNG    6.071260e+05
CNG             4.714808e+05
LPG             2.550000e+05
Name: Discounted amount, dtype: float64
```

```
fuel_groups['Discounted amount'].median().sort_values(ascending=False)
```

```
Fuel Type
Diesel          916000.0
Petrol + CNG    580000.0
Petrol          569000.0
CNG             467000.0
LPG             255000.0
Name: Discounted amount, dtype: float64
```

3. Grouping by Variants:

```
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
pd.set_option('display.width', None)
pd.set_option('display.max_colwidth', -1)

variant_groups['Full amount'].mean().sort_values(ascending=False
```

```
Variant
4X4 AT Titanium                3.120000e+06
4X2 AT Titanium                2.883000e+06
2.4 ZX AT                      2.804000e+06
2.8 4x2 AT                     2.780000e+06
Safari XZA+ ADVENTURE          2.300000e+06
XZ+ 2.0 KRYOTEC                2.294500e+06
TIGUAN HIGHLINE A/T            2.139000e+06
AT GL DIESEL                   2.088000e+06
PLUS SMART 2.0                 2.087000e+06
ZX CVT PETROL                  2.076750e+06
1.4G DCT DUAL                  2.074000e+06
TURBO AT S                     2.066000e+06
Safari XT+2.0 KRYOTEC          2.047333e+06
1.5TSI AT (6                   2.022000e+06
(O) 1.4 TURBO                  2.000000e+06
AT 7 STR                       1.981500e+06
GLS 4WD AT                     1.970000e+06
GDI DCT DUAL                   1.966500e+06
XT 2.0L Kryotec                1.964000e+06
GDI PETROL AT                  1.958286e+06
XZA+ DARK EDITION              1.929000e+06
LIMITED PLUS AT                1.920000e+06
STYLE 1.5 TSI                  1.894000e+06
2.0 SX MT                      1.883000e+06
Camry HYBRID                   1.856000e+06
Scorpio S11                    1.828182e+06
Q3 35TDI PREMIUM               1.815000e+06
PRESTIGE 1.5 MT                1.810000e+06
SX (O) AT                      1.797000e+06
AT GLS PETROL                  1.756000e+06
XUV500 W9 AT                   1.753000e+06
Compass 2.0 LIMITED            1.723000e+06
PLUS 1.4G MT                   1.721500e+06
TOPLINE 1.0 TSI                1.720400e+06
SHARP DCT PETROL               1.715692e+06
XZ 2.0L Kryotec                1.712250e+06
XM 2.0 KRYOTEC                 1.711667e+06
New 2WD MT                     1.707000e+06
STYLE PLUS AT                  1.705000e+06
HT 4WD AT                      1.700600e+06
VX CVT i-VTEC                  1.683000e+06
+ AT PETROL                    1.673583e+06
```

Results

In the used cars marketplace, car prices very up to 31,00,000 rupees depending upon various features such as car brand, model, kilometre driven, registration year, etc. some brands are overpriced as compared to others such as Maruti and Audi. one should avoid buying diesel operated used cars as they are overpriced.

Titanium AT 4 x 4 is the most expensive vehicle in the used cars collection with staggering value of up to 31,00,000. Diesel petrol and petrol plus CNG vehicles have higher values with diesel vehicles being most overpriced. From the data set of used cars Maruti and Hyundai are the most used cars that are sold in the market.

Fiat Mercedes and skoda are the cars that have been sold very less in the used cars marketplace. Mg Audi and Maruti have the highest prices in the used cars marketplace.

As our used cars data set contains lot of outliers predicting used car prices very accurately was a tough objective but by using xg boost that is a boosting technique we have got a very good 99 percent plus accuracy in the training data set with up to 85% accuracy on the test data set.

The average value of a used car from our data set is coming out to be around movie ₹70,00,000 whereas the mean absolute value of error is ₹50,000.

We have also Tested our model by cross validation method in which our model accuracy among different subsets of the data is also uniform that means with such data we can rely on our model and its prediction keeping in mind the error.

CONCLUSION

Using data mining and machine learning approaches, this project proposed a scalable framework for Indian used car market. cars24.com website was scraped using the selenium library to collect the benchmark data. An efficient machine learning model is built by training, testing, and evaluating various regressor models. As a result of pre-processing and transformation, XGBoost Regressor came out on top with 99% accuracy on the training data and 85 % acuuracy on the test data followed by Ensemble Regressor with 84%. Each experiment was performed in real-time within the jupyter environment. In comparison to the system's integrated Jupyter notebook and Anaconda's platform, algorithms took less training time in Google Colab.

Limitations and Future Work

more data reflecting the technical specifications of the cars could have been scrapped and could have been taken into consideration for model building that could have contributed more in tracking the variance of the data set. More feature selection techniques could have been used to select the best features for training our model. With this we can use this model at full scale for predicting used car prices in the Indian marketplace.

In the future, more data will be collected using different web-scraping techniques, and deep learning classifiers will be tested. Algorithms like Quantile Regression, ANN and SVM will be tested. Afterwards, the intelligent model will be integrated with web and mobile-based applications for public use. Moreover, after the data collection phase Semiconductor shortages have incurred after the pandemic which led to an increase in car prices, and greatly affected the secondhand market. Hence having a regular Data collection and analysis is required periodically, ideally, we would be having a real time processing program.