Car Price Prediction Project

Submitted by:

ABHISHEK PAI

ACKNOWLEDGMENT

The background information relating to the project was been provided by fliprobo as a part of the internship phase.

The data was collected from various websites to aid this project.

Related guidance was been provided by fliprobo for the completeion of this project

INTRODUCTION

- Business Problem Framing
  With the covid 19 impact in the market, A lot of changes are seen in the car market. Now some cars are in demand hence making them costly and some are not in demand hence cheaper. client works with small traders, who sell used cars. With the change in market due to covid 19 impact, 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.

- Conceptual Background of the Domain Problem
  To make a machine learning models from new data to do valuate car price.

- Review of Literature
  There is not much research performed as the Data and related information was provided by the source itself, which was been taken into consideration based on the information given by Flip Robo.
- Motivation for the Problem Undertaken
  The Project was assigned by flip Robo as part of the internship phase for better understanding the concept and getting the idea of the industry.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  After importing data various analyses were performed which had univariate, bivariate, and multivariate analysis.

  Univariate analysis: Univariate analysis is the simplest form of analyzing data. It doesn't deal with causes or relationships and its major purpose is to describe; It takes data, summarizes that data, and finds patterns in the data.

  Bivariate analysis: Bivariate analysis is one of the simplest forms of quantitative analysis. It involves the analysis of two variables, to determine the empirical relationship between them. Bivariate analysis can help test simple hypotheses of association.

  Multivariate analysis: Multivariate statistics is a subdivision of statistics encompassing the simultaneous observation and analysis of more than one outcome variable. Multivariate statistics concerns understanding the different aims and backgrounds of each of the different forms of multivariate analysis, and how they relate to each other.

- Data Sources and their formats

After loading the data, the information of data was been checked and a five-row sample was been observed.

- Data Pre-processing Done

  The entire data was in form of CSV and was a mixture of numbers, and objects. The output variable is information in a numerical pattern. The output was based on the data which was provided by a source on the behavioural pattern of the entity. The object part was been converted and extracted to perform ML

- Hardware and Software Requirements and Tools Used

  The system with a 16 core processor was been used,
  The operating system was Windows 10,
  Anaconda 3 was been used for performing ML
  Libraries:

```
import pandas as pd
import selenium
from selenium import webdriver
import time
from selenium.common.exceptions import
StaleElementReferenceException, NoSuchElementException
import urllib
import numpy as np

import re
from pylab import rcParams
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import warnings # Ignores any warning
warnings.filterwarnings("ignore")
import re
from sklearn.metrics import classification_report
from scipy.stats import skew
from sklearn.preprocessing import power_transform
from sklearn import metrics
```

```
from sklearn.impute import SimpleImputer
import xgboost as xgb
from xgboost.sklearn import XGBRegressor
from sklearn.preprocessing import
LabelEncoder,StandardScaler,OneHotEncoder,MinMaxScaler
from sklearn.model_selection import
train_test_split,cross_val_score,cross_val_predict,GridSearchCV
from sklearn.metrics import
accuracy_score,confusion_matrix,f1_score,mean_squared_error as
mse,roc_curve,precision_recall_curve,mean_absolute_error
from sklearn.linear_model import LinearRegression,Ridge,Lasso
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
from sklearn.ensemble import
RandomForestRegressor,AdaBoostRegressor,ExtraTreesRegressor,G
radientBoostingRegressor
from sklearn.metrics import r2_score
from math import sqrt

import statistics
from sklearn.metrics import roc_auc_score
from sklearn.model_selection import cross_val_score

from sklearn.model_selection import train_test_split
```

Model/s Development and Evaluation

.

- Testing of Identified Approaches (Algorithms)

XGBRegressor	GradientBoostingRegressor
RandomForestRegressor	DecisionTreeRegressor
Ridge	LinearRegression
Lasso	ExtraTreesRegressor
AdaBoostRegressor	SVR

.

- Run and evaluate selected models

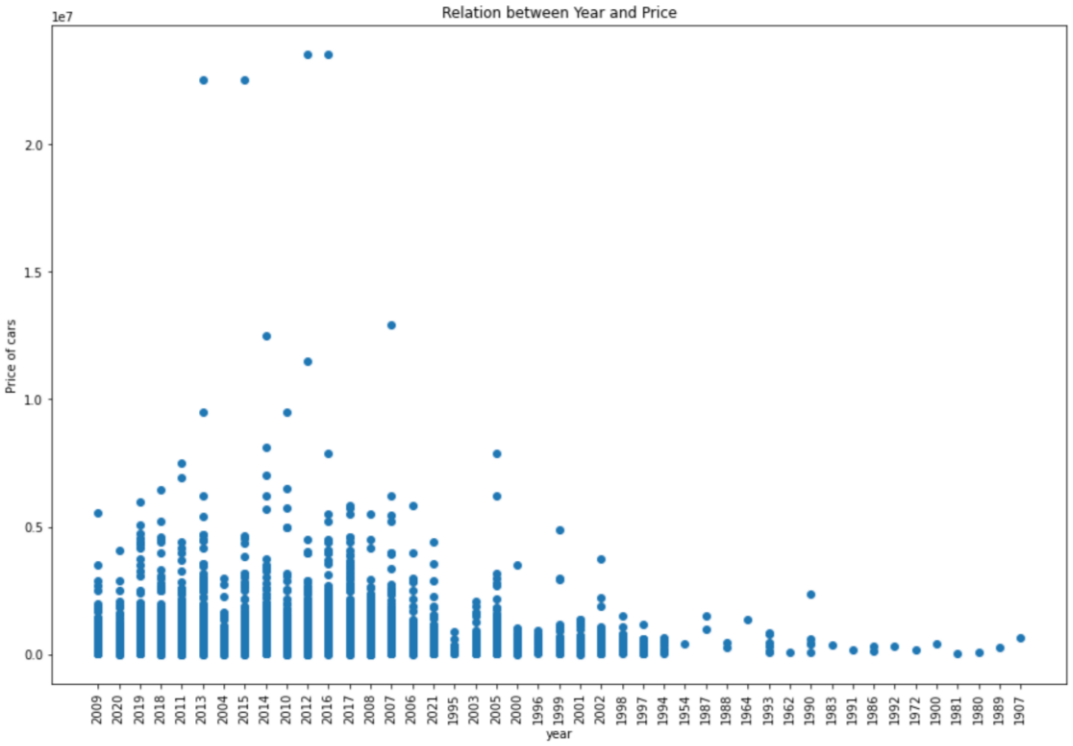
```
xgr:
 Train: 0.715244
Test: -0.096988
MSE: 699707094288.709595


cross val score :  -0.5661293447920587
-cval:   0.5661293447920587

-----------------------------------------------------------------------------------------------------------------
gbr:
 Train: 0.161681
Test: 0.026896
MSE: 620688375880.217163


cross val score :  -0.6012501059552194
-cval:   0.6010265072848202

-----------------------------------------------------------------------------------------------------------------
forest:
 Train: 0.844100
Test: -0.068237
MSE: 681368636603.571411


cross val score :  -0.5805630051465538
-cval:   0.580364367811373

-----------------------------------------------------------------------------------------------------------------
tree:
 Train: 0.997176
Test: -1.324675
MSE: 1482779535553.341309


cross val score :  -0.474478478074656
-cval:   0.47563828101003036
```

```
ridge:
 Train: 0.042385
Test: 0.038640
MSE: 613197289409.978882


cross val score :  -0.5841005348033658
-cval:   0.5841005348033658

-----------------------------------------------------------------------------------------------------------
lin:
 Train: 0.042385
Test: 0.038640
MSE: 613197353536.910278


cross val score :  -0.5838154329679102
-cval:   0.5838154329679102

-----------------------------------------------------------------------------------------------------------
lasso:
 Train: 0.042385
Test: 0.038641
MSE: 613197256101.126953


cross val score :  -0.5838220072188857
-cval:   0.5838220072188857

-----------------------------------------------------------------------------------------------------------
Extra Tree:
 Train: 0.997176
Test: -0.321009
MSE: 842597787890.897217


cross val score :  -0.5708677249754818
-cval:   0.5730089233095575

-----------------------------------------------------------------------------------------------------------
Adaboost:
 Train: 0.114724
Test: -0.188877
MSE: 758317757137.169800


cross val score :  -0.4240037636208471
-cval:   0.4244957628084938

-----------------------------------------------------------------------------------------------------------
Svr:
 Train: -0.043262
Test: -0.041876
MSE: 664554219634.903198


cross val score :  -0.6076523835924462
-cval:   0.6076523835924462
```

- Key Metrics for success in solving problem under consideration

Cross-Validation is **used** to evaluate the performance of a classification model. It is the amount of the variation in the output dependent attributew which is predictable from the input independent variable.

- Visualizations



Overview

Overview | Warnings 5 | Reproduction

Dataset statistics

Number of variables	10
Number of observations	11337
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	885.8 KiB
Average record size in memory	80.0 B

Variable types

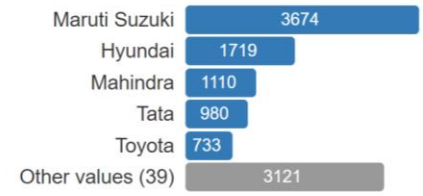
Numeric	4
Categorical	6

Brand

Categorical

HIGH CORRELATION

Distinct	44
Distinct (%)	0.4%
Missing	0
Missing (%)	0.0%
Memory size	88.7 KiB



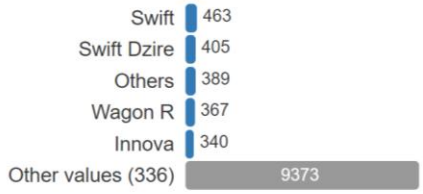
Toggle details

Model

Categorical

HIGH CARDINALITY

Distinct	341
Distinct (%)	3.0%
Missing	0
Missing (%)	0.0%
Memory size	88.7 KiB



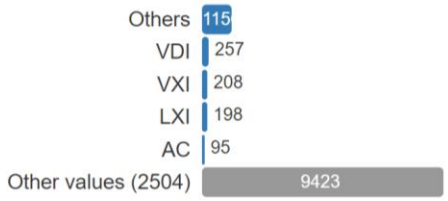
Toggle details

Variant

Categorical

HIGH CARDINALITY

Distinct	2509
Distinct (%)	22.1%
Missing	0
Missing (%)	0.0%
Memory size	88.7 KiB



Toggle details

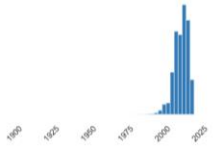
Year

Real number ($\mathbb{R}_{\geq 0}$)

HIGH CORRELATION

Distinct	45
Distinct (%)	0.4%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	2012.198377

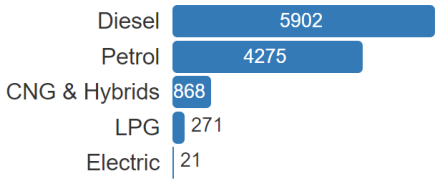
Minimum	1900
Maximum	2021
Zeros	0
Zeros (%)	0.0%
Negative	0
Negative (%)	0.0%
Memory size	88.7 KiB



Toggle details

Fuel
Categorical

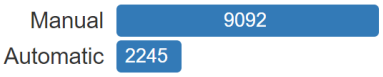
Distinct	5
Distinct (%)	< 0.1%
Missing	0
Missing (%)	0.0%
Memory size	88.7 KiB



Toggle details

Transmission
Categorical

Distinct	2
Distinct (%)	< 0.1%
Missing	0
Missing (%)	0.0%
Memory size	88.7 KiB

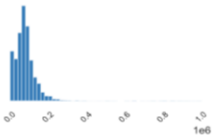


Toggle details

KM_driven
Real number ($\mathbb{R}_{\geq 0}$)

Distinct	1997
Distinct (%)	17.6%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	79704.819

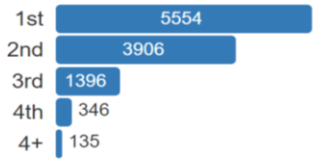
Minimum	0
Maximum	999990
Zeros	59
Zeros (%)	0.5%
Negative	0
Negative (%)	0.0%
Memory size	88.7 KiB



Toggle details

No_of_Owners
Categorical

Distinct	5
Distinct (%)	< 0.1%
Missing	0
Missing (%)	0.0%
Memory size	88.7 KiB



Toggle details

price

Real number ($\mathbb{R}_{\geq 0}$)

Distinct	994
Distinct (%)	8.8%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	502013.4059

Minimum	15000
Maximum	23500000
Zeros	0
Zeros (%)	0.0%
Negative	0
Negative (%)	0.0%
Memory size	88.7 KiB

Toggle details

- Interpretation of the Results

  1 model has been used

  The random forest has performed better after grid search cv.

  The finalized model is Random Forest.

  .

CONCLUSION

- Key Findings and Conclusions of the Study

  As cross-validation score was considered for evaluating the models

  The score for GBR before tuning was 0.6

  After tuning the score was 0.48 which was been reduced.

  Considering the score the best way to improve the result would be adding more data to dataset

- Learning Outcomes of the Study in respect of Data Science

  Adding more data can help to increase the accuracy.

- Limitations of this work and Scope for Future Work

  There is a lot of scopes, more tweaks in a model can help to get better results.