Pune Institute of Computer Engineering
Dhankawadi, Pune

A MINI-PROJECT REPORT

ON

“Car Sales’ Price Prediction”

SUBMITTED BY

Name: Mihir Virendra Parte

Roll No: 31350

Class: TE-3

Under the guidance of

Prof. Priyanka N. Savadekar

for

Honours* in Data Science

Data Science and Visualization (310501)

DEPARTMENT OF COMPUTER ENGINEERING

Academic Year 2022-23

Table of Contents

1. INTRODUCTION

In the world of business, it is important to know which products are profitable for the company or distributor and which products are leading to a loss, which features of the product attract the customer more or which features dissuade them from purchasing the product. To learn the best and worst features of a product, it is important to analyse their performances over a period of time to make a conclusion about the product.

I have chosen to work on analysing trends in car sales and predicting future prices of those cars based on these trends, as I am passionate about cars. I have taken the dataset from a reliable source Kaggle and pre-processed it (Checked and replaced outliers, missing values and redundant data), analysed feature importance of the dataset i.e., choose features that will help increase accuracy of predictions, train the prediction model and predict car sales with this model.

2. MOTIVATION

Many car distributors fail in their businesses quickly because they do not do a complete market survey and analysis of the previous performances of the products. In India, many businesses have been popping up with owners having no prior experience or proof of expertise in the field. Their primary reason for opening up those businesses is seeing other businesses gaining popularity.

I aim to provide new businesses a way to decide which products to choose, in order to maximize profits and reduce expenditure loss and predicting future sales based on viable features decided earlier.

3. LITERATURE SURVEY

I gathered and analysed information on the following topics:

3.1 Supervised Learning

It is a machine learning paradigm that relies on labelled data and can help to predict outcomes for future datasets. The two most common supervised learning techniques are '**Regression**' and '**Classification**'.

As I am working on a labelled dataset and the variable to predict i.e., MSRP which is a collection of continuous values and regression is perfectly viable for this prediction as regression is used to predict continuous values. Thus, I have chosen to use supervised regression techniques for my prediction.

3.2 Regression

It is a supervised learning technique used to relate a dependent variable to one or more independent variables. There are various types of regression techniques like 'Linear', 'Lasso' and 'Ridge'. Random forests are capable of performing both regression and classification techniques.

3.3 Linear Regression

It is a linear approach for identifying a relationship between a one dependent variable and one or more independent variables.

The equation of multiple linear regression is as follows:

$$y = b_1x_1 + b_2x_2 + \ldots + b_n x_n + c$$

where

- $b_1, b_2, \ldots b_n$ are the regression coefficients
- $x_1, x_2, \ldots. x_n$ are the independent variables
- c is a constant and
- y is the dependent variable

Multiple linear regression will be used on this dataset as there are many independent variables and one dependent variable i.e., MSRP.

3.4 Random Forest

In simple terms, it is a **forest of decision trees**. As a decision tree combines all decisions to come to conclusion, a random forest combines all the available decision trees, and its final output will the **average of the outputs of all of the decision trees**. Random forests are quite flexible as they can be used for regression and classification tasks. It can **handle**

large datasets very efficiently. It also maintains accuracy when a large proportion of the data is missing.

Random forest will be used on this dataset as there are many independent variables and we need our prediction model to be as accurate as possible.

3.5 r^2 Score

In ML, it is a measure of goodness of fitting of a model. In statistical analysis, the r^2 coefficient of determination is a measure of **how well the regression predictions approximate the real data points**.

r^2 = proportion of the variation in the dependent variable that is predictable from the independent variables.

Research Paper References:

Title	Analysis of linear regression on used car sales in Indonesia
Authors	C K Puteri and L N Safitri
Month and Year of Publication	August 2018
Link	https://iopscience.iop.org/article/10.1088/1742-6596/1469/1/012143/pdf
Summary	• Used multiple linear regression to predict car sales. • Their experimentation proved that more variables led to higher accuracy in predicting prices. • The experiment got an accuracy value above 75% when variables age, distance, colour of car, transmission and cities of car sales were combined and used.

Title	Comparative Analysis of Car Sales Using Supervised Algorithms
Authors	Prashant Gupta, Pradumn Kumar, Kundan Kumar and Nidhi Singh

Month and Year of Publication	January 2021
Link	https://www.mililink.com/upload/article/1776193310aams_vol_203_january_2020_a4_p367-375_prashant_gupta_and_nidhi_singh.pdf
Summary	• The authors made the predictions using linear regression, decision trees and random forest. • Linear regression provided the best accuracy among the 3 for the dataset used.

4. PROBLEM DEFINITION

In the world of business, it is important to know which products are profitable for the company or distributor and which products are leading to a loss, which features of the product attract the customer more or which features dissuade them from purchasing the product. To learn the best and worst features of a product, it is important to analyse their performances over a period of time to make a conclusion about the product.

Analyse trends in car sales and **predict future car prices (MSRP)** using different car features as parameters for prediction.

5. METHODOLOGY

5.1 Methodology used

1. <u>Problem Understanding</u>: Understanding the problem statement and identifying the different dependent and independent variables in the dataset.

2. <u>Data Collection</u>: Data can be collected and categorized using data scraping and mining or the data can be taken from an authentic source like Kaggle. For the given problem definition, I have taken a dataset from Kaggle.

1	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	city mpg	Popularity	MSRP
2	BMW	1 Series M	2011	premium unleade	335	6	MANUAL	rear wheel drive	2	Factory Tuner,Luxury,	Compact	Coupe	26	19	3916	46135
3	BMW	1 Series	2011	premium unleade	300	6	MANUAL	rear wheel drive	2	Luxury,Performance	Compact	Convertible	28	19	3916	40650
19	Audi	100	1992	regular unleaded	172	6	MANUAL	front wheel drive	4	Luxury	Midsize	Sedan	24	17	3105	2000
20	Audi	100	1992	regular unleaded	172	6	MANUAL	front wheel drive	4	Luxury	Midsize	Sedan	24	17	3105	2000
34	FIAT	124 Spide	2017	premium unleade	160	4	MANUAL	rear wheel drive	2	Performance	Compact	Convertible	35	26	819	27495
37	Mercedes	190-Class	1991	regular unleaded	130	4	MANUAL	rear wheel drive	4	Luxury	Compact	Sedan	26	18	617	2000
66	Chrysler	200	2015	flex-fuel (unleade	184	4	AUTOMATIC	front wheel drive	4	Flex Fuel	Midsize	Sedan	36	23	1013	25170
67	Chrysler	200	2015	flex-fuel (unleade	184	4	AUTOMATIC	front wheel drive	4	Flex Fuel	Midsize	Sedan	36	23	1013	23950
89	Nissan	200SX	1996	regular unleaded	115	4	MANUAL	front wheel drive	2	N/A	Compact	Coupe	36	26	2009	2000
90	Nissan	200SX	1996	regular unleaded	115	4	MANUAL	front wheel drive	2	N/A	Compact	Coupe	36	26	2009	2000

Glimpse of the Dataset

3. <u>Importing Necessary Libraries</u>: To process data and create a prediction model, we need to import the necessary Python libraries.
   A. Numpy - For dealing with arrays.
   B. Pandas - For reading dataset, manipulating data columns and analysis.
   C. Matplotlib, Seaborn, Plotly - For analysis through data visualization.
   D. Sklearn - For using machine learning techniques to train/test prediction models, replacing missing values, encoding categorical features and scaling values.

4. <u>Data Preparation</u>:

   A. Dropping duplicate rows in dataset.

Finding number of duplicate values

```
print('Number of duplicates are : ', dataset.duplicated().sum())
dataset.shape
```

✓ 0.1s

```
Number of duplicates are :  715

(11914, 16)
```

Dropping duplicate values

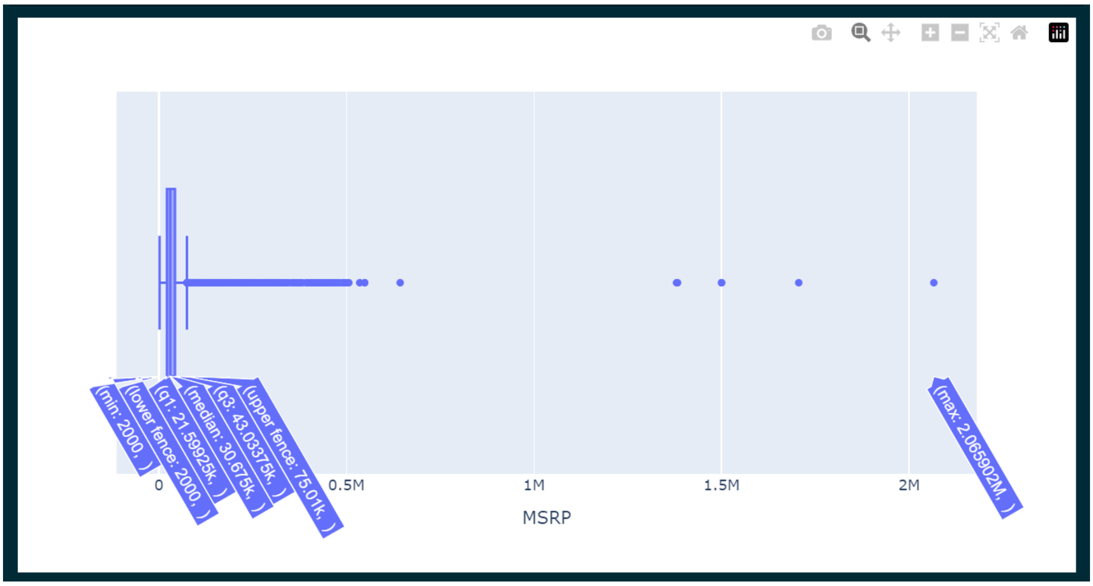
```
dataset = dataset.drop_duplicates()
print('After removing duplicates, number of duplicates: ', dataset.duplicated().sum())
dataset.shape
```

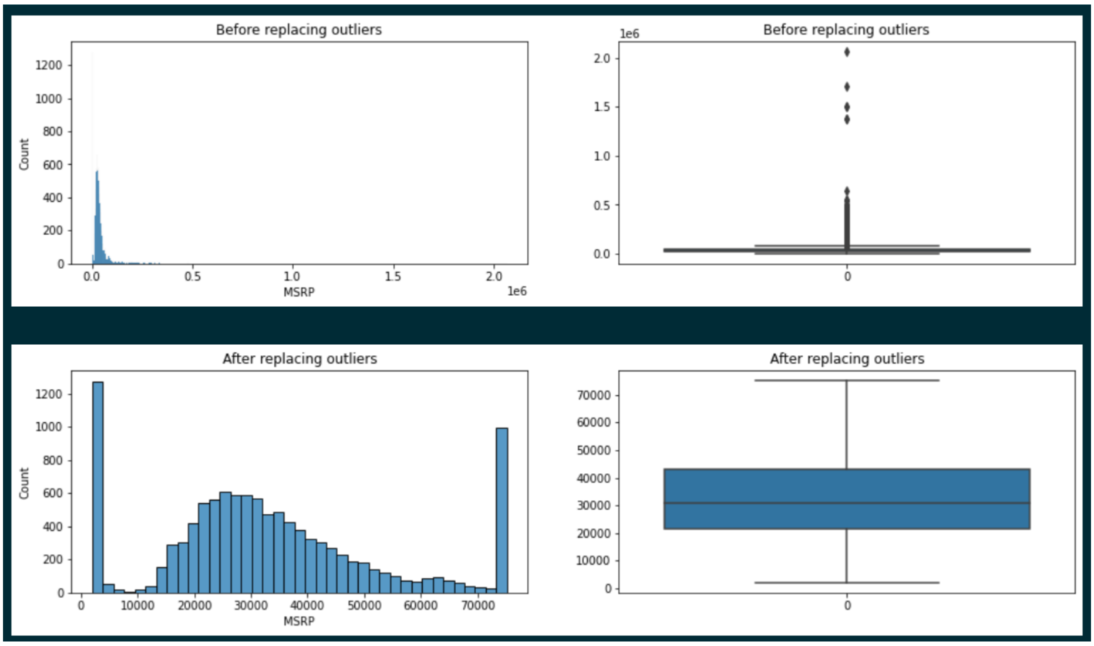
✓ 0.1s

```
After removing duplicates, number of duplicates:  0

(11199, 16)
```

B. Replacing missing data and outliers.





C. Splitting dataset into independent variable set and dependent variable set.

D. Encoding categorical features.

```
ct = ColumnTransformer(transformers=[('encoder',OrdinalEncoder(encoded_missing_value=-1),
[0,1,3,6,7,8,9,10,11])],remainder='passthrough')
X = np.array(ct.fit_transform(X))
print(X[0])
```

✓ 0.1s

```
[4.0 1.0 8.0 3.0 3.0 0.0 38.0 0.0 8.0 2011 335.0 6.0 26 19 3916]
```

E. Scaling large values to smaller values.

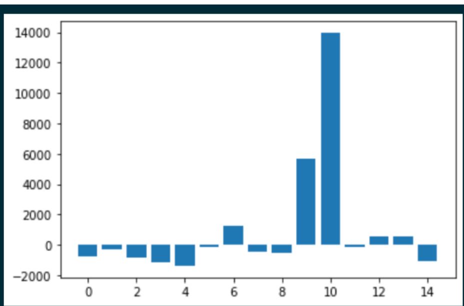
```
sc = StandardScaler()
X[:,0:] = sc.fit_transform(X[:,0:])
print(X[0])
```

✓ 0.1s

```
[-1.3119569392803285 -1.693869250160844 0.01876595395717283
 1.732665771724385 1.149733936180923 -1.6664466914795872
 0.33563270705056925 -1.1191415063564727 -0.07585878587365144
 0.0394958931874387 0.74568542006128 0.18841862482373536
 -0.06801536273712086 -0.07974713577472532 1.6308173758635829]
```

F. Choosing viable features for prediction.

```
Feature: 0, Score: -801.55593
Feature: 1, Score: -283.51129
Feature: 2, Score: -882.57020
Feature: 3, Score: -1189.46061
Feature: 4, Score: -1386.07463
Feature: 5, Score: -163.80122
Feature: 6, Score: 1235.53181
Feature: 7, Score: -481.30109
Feature: 8, Score: -571.65429
Feature: 9, Score: 5653.98198
Feature: 10, Score: 13952.53185
Feature: 11, Score: -135.47611
Feature: 12, Score: 537.70884
Feature: 13, Score: 583.63896
Feature: 14, Score: -1053.45533
```



This output indicates that features 6, 9,10, 12, 13 are the most viable features for prediction

i.e Features-'Market Category','Year', 'Engine HP', 'Highway MPG' and 'City MPG' respectively

G. Splitting independent and dependent variable sets into training and test sets.

5. Modeling:
   A. Fitting training sets into different regression models and checking their r^2 accuracies.

```python
models = {
    "Linear regression": LinearRegression(),
    "Linear regression (Ridge)": Ridge(),
    "Linear regression (Lasso)": Lasso(),
    "Random forest": RandomForestRegressor(),
    "Gradient boosting": GradientBoostingRegressor()
}

#Fitting training data into models and checking each model's accuracy
for item, model in models.items():
    model.fit(X_train, y_train)
for name, model in models.items():
    print(name + ' ' + 'R^2 Score: {:.3f}'.format(model.score(X_test, y_test)))
```

✓ 3.9s

```
Linear regression R^2 Score: 0.766
Linear regression (Ridge) R^2 Score: 0.766
Linear regression (Lasso) R^2 Score: 0.766
Random forest R^2 Score: 0.957
Gradient boosting R^2 Score: 0.913
```

   B. Choosing best models from the step A to start building them for predictions.
   C. Fitting training sets into the chosen models (Random Forest) i.e., ‘X_train’ and ‘y_train’.

```python
rf_regressor = RandomForestRegressor()
rf_regressor.fit(X_train,y_train)
```

✓ 3.2s

```
▾ RandomForestRegressor
RandomForestRegressor()
```

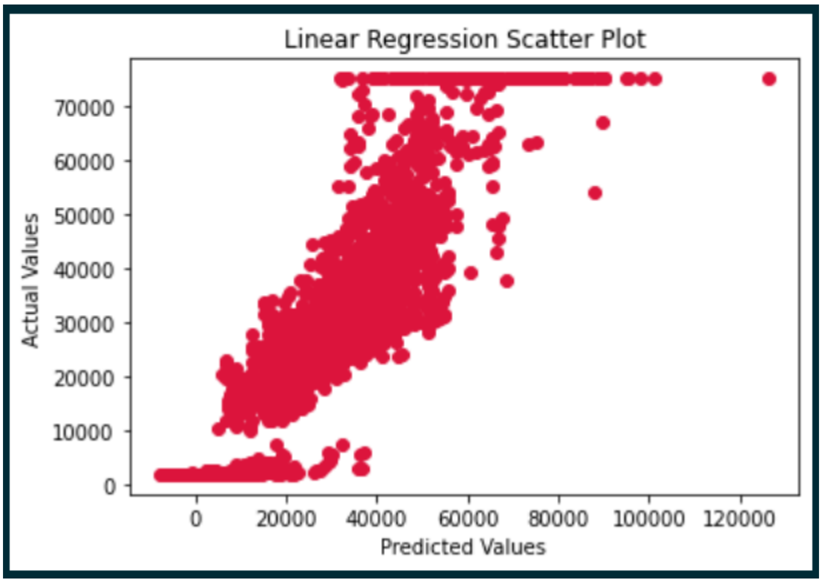
6. Evaluation/Prediction:
   A. Predicting dependent variable using independent training set i.e., ‘X_test’.
   B. Comparing prediction results with dependent test set i.e., ‘y_test’.

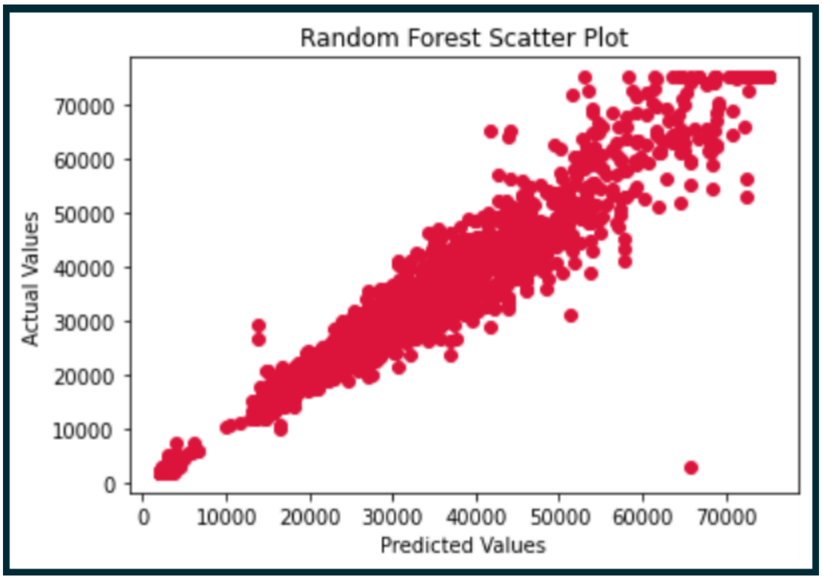
```
      Predicted      Actual                   Predicted      Actual
0     39023.356923   42600.0            0     44131.816667   42600.0
1     24874.943599   21795.0            1     21821.850000   21795.0
2      5311.877618    2350.0            2      2782.294762    2350.0
3     77291.030784   75182.0            3     75182.000000   75182.0
4     27541.162991   28200.0            4     30935.202103   28200.0
...            ...       ...            ...            ...       ...
2235  31552.695116   33540.0            2235  29249.272500   33540.0
2236  16869.480835   16580.0            2236  15518.841667   16580.0
2237  42807.965767   38215.0            2237  35496.948375   38215.0
2238  28713.950508   25995.0            2238  27593.001587   25995.0
2239  18842.401565   22305.0            2239  21727.647500   22305.0

[2240 rows x 2 columns]                 [2240 rows x 2 columns]

Linear Regression Accuracy= 0.766       Random Forest Accuracy= 0.955
```

C. Plotting scatter plots for chosen regression models to compare prediction results.





D. Comparing r^2 scores of the chosen model and drawing a conclusion.

5.2 Algorithms used

1. For finding best algorithm, I used following algorithms for comparison.
   a. Multiple Linear Regression
   b. Lasso Regression
   c. Ridge Regression
   d. Gradient Boosting Regression
   e. Random Forest Regression
2. After selecting best algorithms, following algorithms were used for building prediction models:
   a. Multiple Linear Regression (For comparison)
   b. Random Forest Regression (Best r^2 score)

5.3 Dataset used

Name: ‘Car Features and MSRP’

Source: Kaggle

Link: https://www.kaggle.com/datasets/CooperUnion/cardataset?datasetId=575

Author: @cooperunion on Kaggle

6. CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

During the experimentation of comparing the r^2 scores, results were drawn.

Regression Algorithm	R^2 Score
Multiple Linear	0.766
Lasso	0.766
Ridge	0.766
Gradient Boosting	0.913
Random Forest	0.958

Hence, random forest gave the best r^2 score and was paired with multiple linear regression (For comparative analysis) to form two different prediction models.

The random forest model did well in predicting very close to the actual values.

Thus, for very large datasets and several features, random forest regression is the most suitable algorithm for predicting future car prices.

We also learned about the car features that were more suitable to analyse for selling the cars i.e., ‘Market Category’, ‘Year’, ‘Engine HP’, Highway MPG’ and ‘City MPG’.

6.2 Future Scope

This experiment has proved that we can gather features of cars that are more attractive to the customer and features that are less important to work on currently. We’ve also predicted future prices based on these features.

Any new and upcoming distributors can use this info to start focusing on the best features of the product and best price to sell them at.

This experiment can be done for different products, not just cars, if given the sufficient history of the products. This can help many new business owners when they’re starting up with products that have already been in the market for a long time.

7. REFERENCES

Dataset: https://www.kaggle.com/datasets/CooperUnion/cardataset?datasetId=575

Research papers:

https://www.mililink.com/upload/article/1776193310aams_vol_203_january_2020_a4_p367-375_prashant_gupta_and_nidhi_singh.pdf

https://iopscience.iop.org/article/10.1088/1742-6596/1469/1/012143/pdf

Python Libraries' Documentations:

https://scikit-learn.org/stable/supervised_learning.html

https://pandas.pydata.org/docs/user_guide/index.html#user-guide

https://pandas.pydata.org/docs/reference/index.html#api

https://numpy.org/doc/stable/reference/index.html#reference

https://plotly.com/python-api-reference/

https://seaborn.pydata.org/api.html

https://matplotlib.org/stable/api/index.html