Analysis on Used Car Dataset Using Regression Algorithms

Sumit Kumar(212CD028)

January 10, 2022

Abstract

In this report, different regression technique is used to determine the factors affecting used car prices in United Kingdom. By using a unique data set on second-hand car prices for 7632 cars(data-split= 65 %) , the regression models were estimated by employing the regression algorithms like Linear Regression, Decision Tree Regression and Random forest Regression. According to the results, while some characteristics of cars such as model, year, transmission, fueltype, miles per gallon, engine size have a positive impact on the regression model, while mileage and tax have a negative impact on the regression model.

1 Introduction

Car Market is on the boom these days, in fact we can observe this from the new model and manufactures are trying to make pavement for themselves in the market. It provides a lot of employment and a major contribution to the national income.

Since the inception of the cars in 1886 and mass production by Ford Motor Company in 1908, car market had grown many folds and will in future. With more than 40 Car Manufactures and 200 different models of the cars and increasing day by day, used car market also has its importance. Before buying a car, customer often looks into the **resale value** of the car, so that in future, they can get a good resale price of the car in the used car market.

In my report, I have worked on the regression problem i,e Analysis on the used car data set with dependent variable to be the price of the vehicle. Motivation behind this problem is when a customer sells his/her car in used car market then they must get a good price according to the specification of the car like transmission, fuel type etc. So this analysis plays a very important role in the determination of the price of the car.

2 Dataset and its Source

Used Car dataset used in the problem is the 100K United Kingdom Used Car dataset. It consists car data from different manufactures like Audi, BMW, Hyundai, skoda, toyato. The Data has been cleaned previously for the competition format. Data from different manufactures are being combined and regression can be applied for the price attributes as dependent variable from the dataset.

Doing the further investigation regarding the data that is begin collected from 2009 to 2019. And the country is United Kingdom where the data is being collected.

3 Attributes in Dataset and its type

Data can be found at this Link :- Dataset.

Dataset contains 10 different independent and 1 dependent variable i.e. price. Shape of the dataset are 4960 rows X 8 Columns.

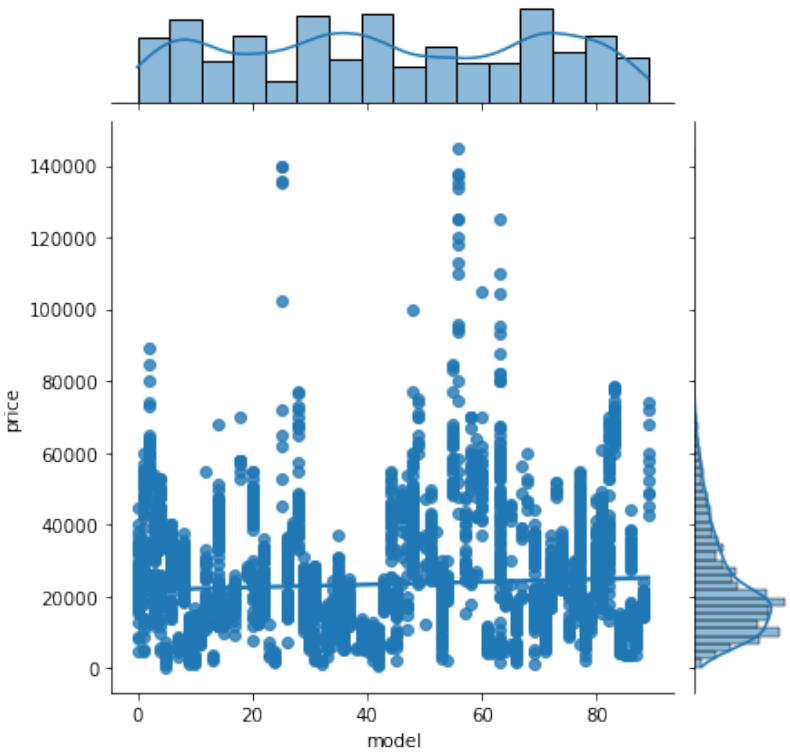
Originally, Dataset contain 10 independent variables and 1 dependent variable:-

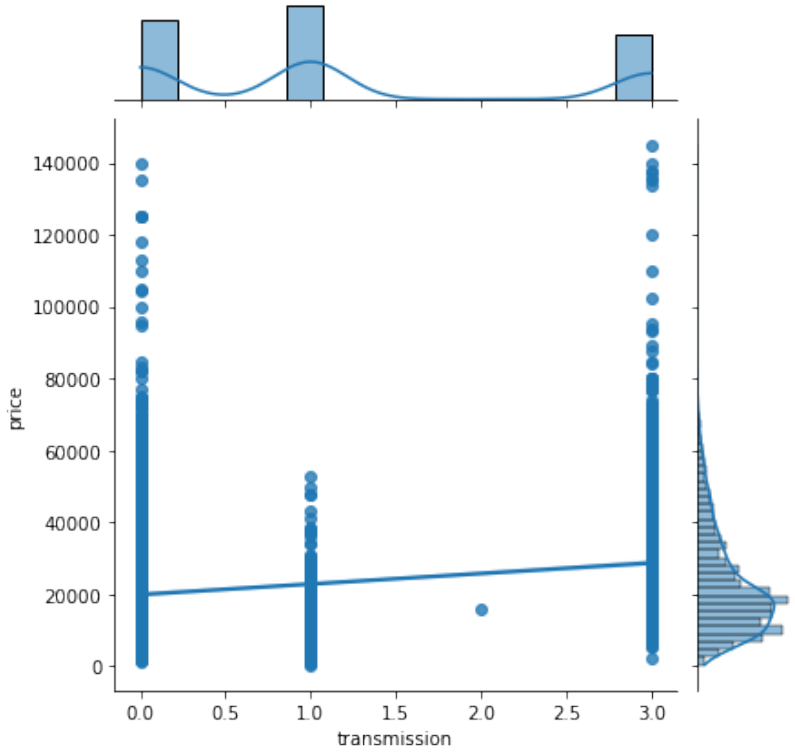
Serial No.	Attributes	Datatype
1	CarID	Integer
2	Brand	String
3	Model	String
4	Year DZ	Integer
5	Transmission	String
6	Mileage	Integer
7	Fueltpye	String
8	Tax	Float
9	Mpg	Float
10	EngineSize	Float

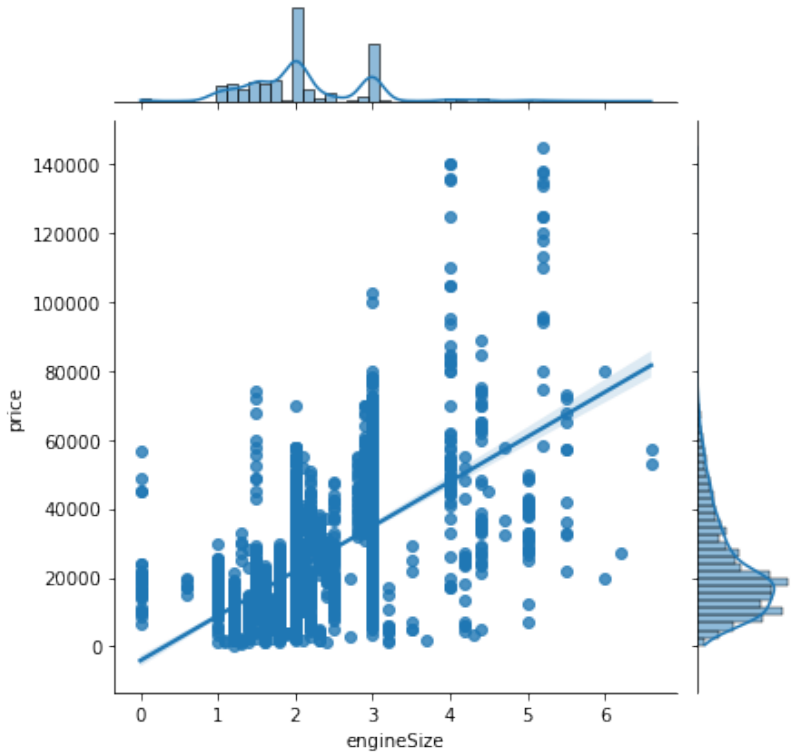
Table 1: Attributes List

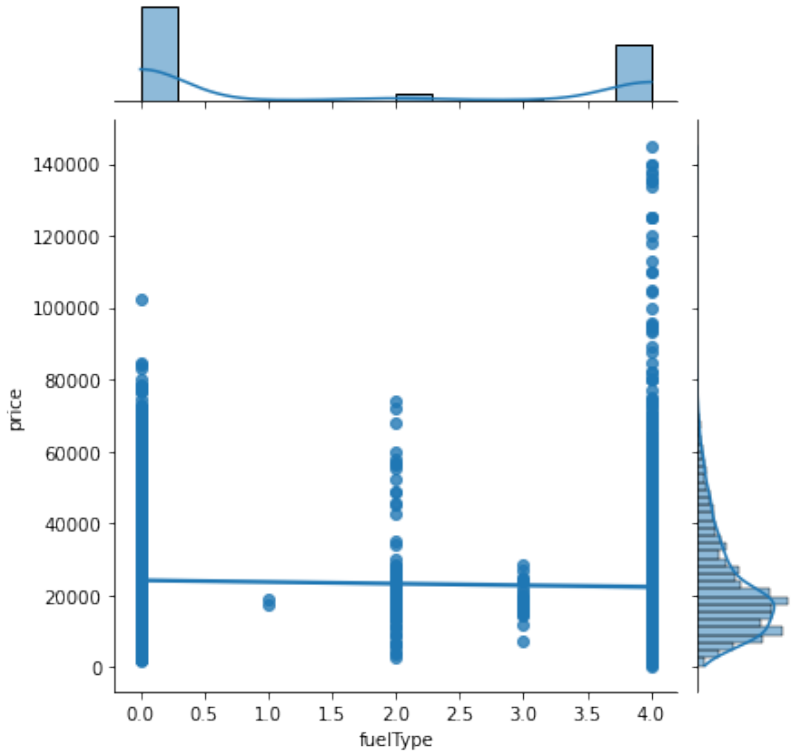
In the original dataset, there were 10 independent variables but while implementing different regression algorithms.

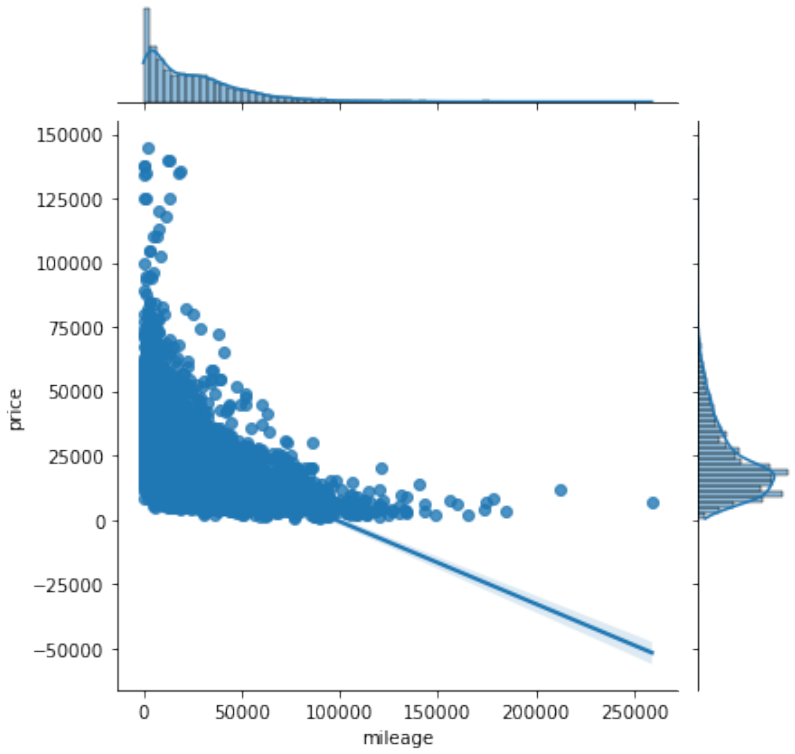
4 Analysis on variables

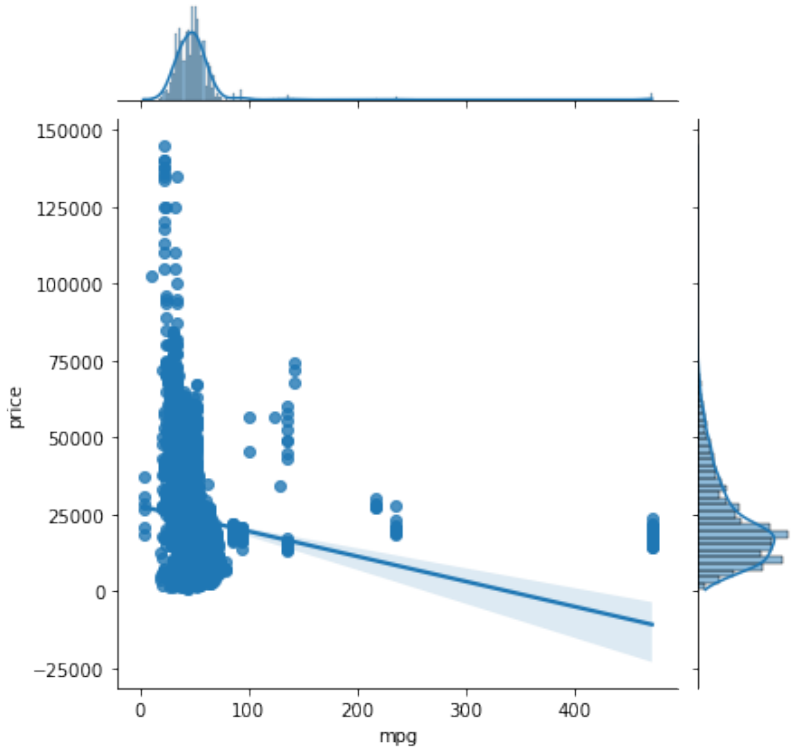


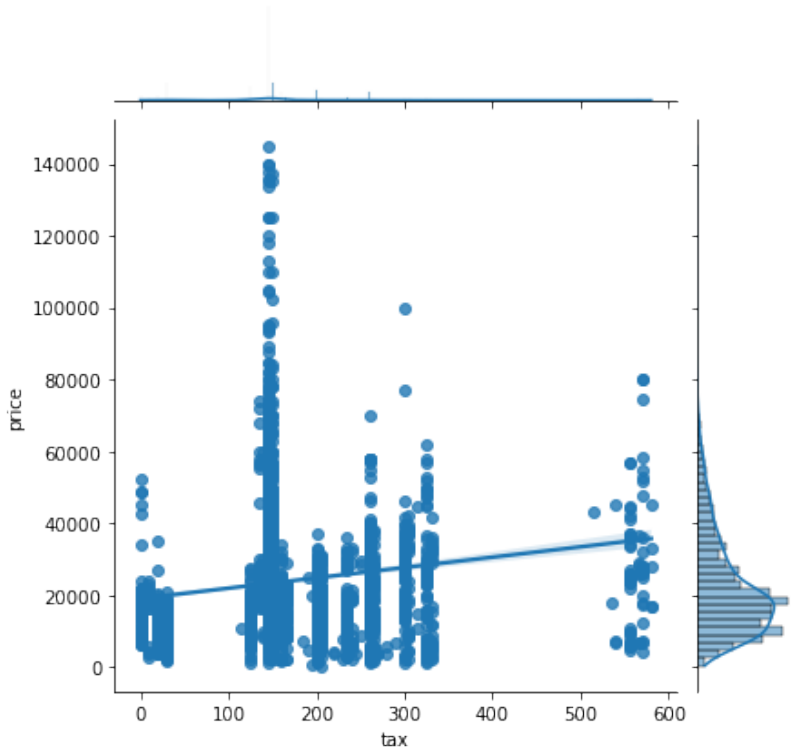

price
transmission
0
20000
40000
60000
80000
100000
120000
140000
0.0
0.5
1.0
1.5
2.0
2.5
3.0

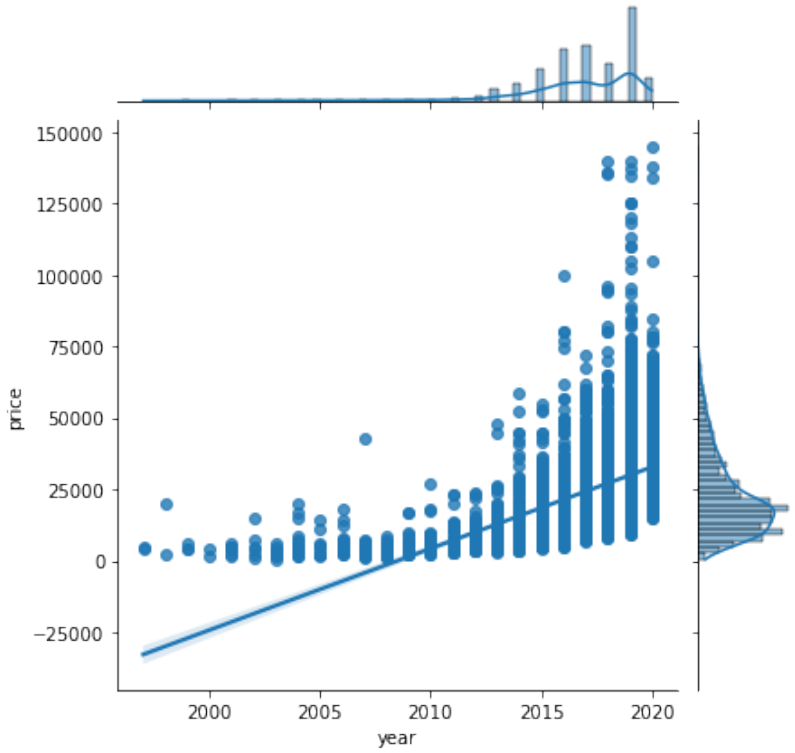

price
engineSize
0
20000
40000
60000
80000
100000
120000
140000
0
1
2
3
4
5
6


price
fuelType
0
20000
40000
60000
80000
100000
120000
140000
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0


price
mileage
−50000
−25000
0
25000
50000
75000
100000
125000
150000
0
50000
100000
150000
200000
250000


150000
125000
100000
75000
50000
25000
0
−25000
price
0
100
200
300
400
mpg


140000
120000
100000
80000
60000
40000
20000
0
price
0
100
200
300
400
500
600
tax



5 Problem Statement and Target Customers

Analysis on the Used Car Data set using the regression algorithms like Linear Regression, Random Forest Regression and Decision Tree Regression provides a accurate prediction using the car specification like mpg, transmission, year, model, tax, mileage etc for the price in the market.

Target customers for this analysis are :-

1. Buyer :- Can predict the price of the car before buying accourding to the specs.
2. Seller :- Can predict the price of the car before selling accourding to the specs.
3. Agent :- Can predict the price of the car before buying accourding to the specs and can get a good price for the car during resale.

6 Data Preprocessing

1. Initially,Data contains 10 independent variables, for better results of the model,I have dropped two variables for the model i.e. carID and Manufacturer.
2. **Standard Scaling** :- Standardization scales each input variable separately by subtracting the mean (called centering) and dividing by the standard deviation to shift the distribution to have a mean of zero and a standard deviation of one.

   A value is standardized as follows:

$$y = \frac{x - mean}{standard\ deviation}$$

where standard deviation is written as :-

$$standard\,deviation = \sqrt{\frac{\sum(x - mean)^2}{count(x)}}$$

This mathematical expression represents the background calculation of the standard scaler in the library scikit-learn

3. Now Data is splited with the test size of 0.2.

7 Concepts Used

7.1 Linear Regression

Because of its straightforward representation, linear regression is a popular model.

The representation is a linear equation that combines a collection of input values (x), with the solution being the projected output for that set of input values (y). As a result, both the input (x) and output (y) values are numeric.

Each input value or column is assigned one scale factor, referred to as a coefficient and denoted by the capital Greek letter Beta in the linear equation (B). One more coefficient is added, which gives the line an extra degree of freedom (for example, going up and down on a two-dimensional plot) and is known as the intercept or bias coefficient.

In a simple regression problem (a single x and a single y), the form of the model would be:

$$y = B_0 + B_1 * x_1 + B_2 * x_2 + B_3 * x_3 + B_4 * x_4 \ .....$$

where B_0 is the bias of the model.

When there are several inputs (x) in higher dimensions, the line is called a plane or a hyper-plane. As a result, the representation is the equation's form as well as the coefficients' specific values (e.g. B0 and B1 in the above example).

Cost Function of the Linear Regression Model is given as below:-

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=0}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

where m is the no of instances on the model, θ_0 *and* θ_1 *are the parameter of the models.*

7.2 Regularisation

It is one of the most essential machine learning principles. By providing new information to the model, this strategy prevents it from overfitting.

It is a type of regression in which the coefficient estimates are reduced to zero. To avoid the problem of overfitting, this strategy forces us not to develop a more sophisticated or flexible model.

A conventional least squares model has some variation, which means it won't generalise well to data sets that aren't the same as the training data. Regularization greatly reduces the model's variance without significantly increasing its bias. As a result, the tuning parameter determines the impact on bias and variance in the regularisation procedures discussed above.

As the value of decreases, the value of coefficients decreases, lowering the variance. This increase in is useful up to a degree because it merely reduces variance (thus avoiding overfitting) without sacrificing any significant characteristics in the data. However, after a certain value, the model begins to lose crucial properties, resulting in bias and underfitting.

7.3 Feature Importance

Feature Importance refers to methods for calculating a score for each of a model's input features; the scores simply describe the "importance" of each feature.

A higher score indicates that a certain feature will have a greater impact on the model used to forecast a given variable.

Feature Importance is extremely useful for the following reasons:

1. Data understanding
2. Model Improvement
3. Model Interpretability

However in my model feature importance analysis shows the following results:-

1. Feature: 0, Score: 1607.72234
2. Feature: 1, Score: 4795.33066
3. Feature: 2, Score: 489.40376
4. Feature: 3, Score: -3769.33351
5. Feature: 4, Score: 757.47698
6. Feature: 5, Score: -1063.27324
7. Feature: 6, Score: 966.04659
8. Feature: 7, Score: 11098.66900

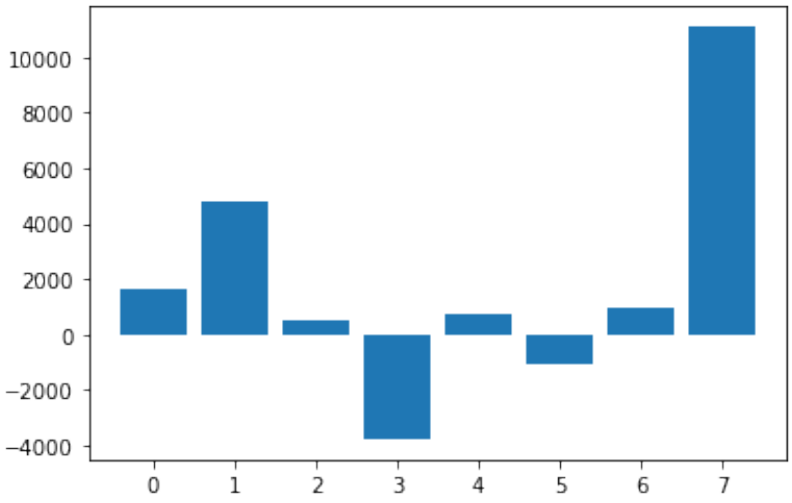


Figure 1: Feature Importance Graph

From the above graph we can see that, we have some features whose feature score is negative which means these features increase the error in the prediction where as the positive score decreases the error and thus improves the model.

7.4 Gradient descent Algorithms

Gradient descent is an iterative optimization approach for determining a function's local minimum.

To use gradient descent to discover a function's local minimum, we must take steps proportional to the negative of the function's gradient (move away from the gradient) at the present location. Gradient Ascent is the process of approaching a local maximum of the function by taking steps proportional to the positive of the gradient (going towards the gradient).

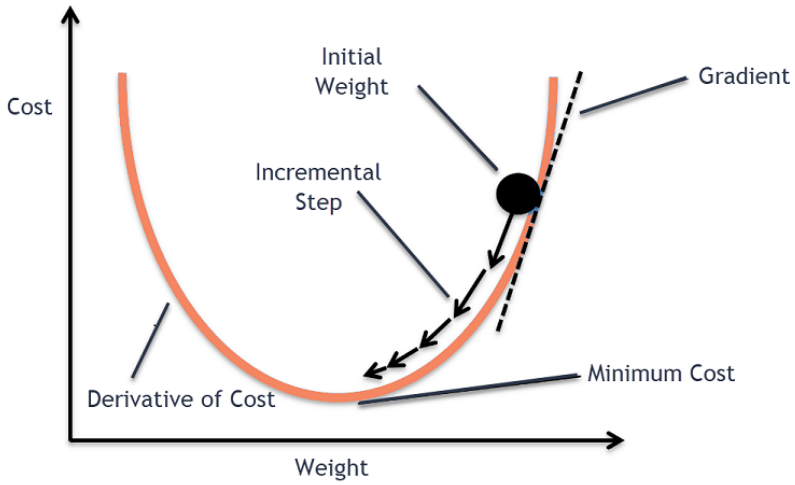


Figure 2: Gradient Descent

Mathematically, we can write the algorithm for the Gradient Descent as follows:-

$$
\begin{aligned}
&repeat\ until\ convergence\{ \\
&\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \\
&\}
\end{aligned}
$$

where α is called Learning rate – a tuning parameter in the optimization process. It decides the length of the steps.

So, we can deduce that from the α that:-

1. Learning rate is optimal, model converges to the minimum
2. Learning rate is too small, it takes more time but converges to the minimum
3. Learning rate is higher than the optimal value, it overshoots but converges ($1/C < eta < 2/C$)
4. Learning rate is very large, it overshoots and diverges, moves away from the minima, performance decreases on learning

now we can see the 3-D Flot of the Gradient Descent Plot:-

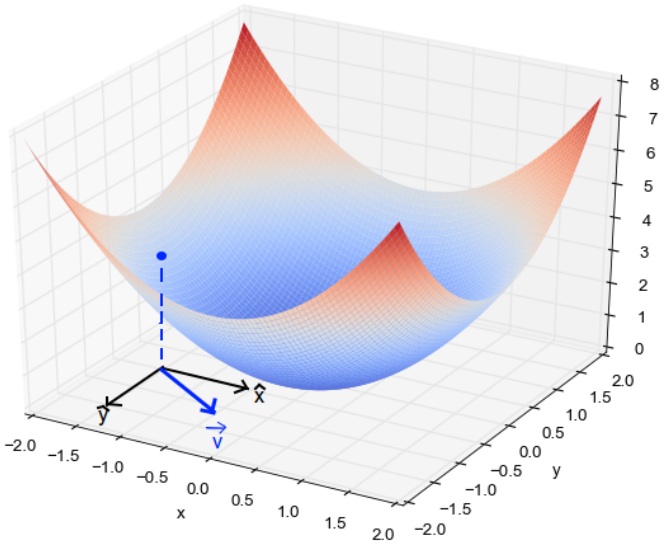


Figure 3: Contour Plot of Gradient Descent

Here are some commanly used Gradient Descent Algorithms are:-

1. Batch Gradient Descent
2. Stochastic Gradient Descent
3. Mini-Batch Gradient Descent

7.5 Random Forest Regression

Because the Decision Tree is a simple to understand and interpret approach, a single tree may not be sufficient for the model to learn the features. Random Forest, on the other hand, is a "Tree"-based algorithm that makes decisions based on the qualitative aspects of many Decision Trees.
As a result, it can be called a "Forest" of trees, hence the name "Random Forest." The name 'Random' comes from the fact that this algorithm is made up of a forest of 'Randomly constructed Decision Trees.'

A random forest is a meta-estimator (that is, it integrates the results of numerous forecasts) that aggregates many decision trees with a few useful modifications: At each node, the amount of characteristics that can be divided on is limited to a certain fraction of the total (which is known as the hyper parameter).
This guarantees that the ensemble model does not rely too strongly on any single feature and that all potentially predictive information are considered equally. When generating splits, each tree takes a random sample from the original data set, adding a layer of unpredictability that inhibits over fitting.

7.6 Decision Tree Regression

One of the most widely used and useful models for supervised learning is the Decision Tree. It can be used to tackle both regression and classification problems, albeit the latter is more widely utilised.

There are three sorts of nodes in this tree-structured classifier. The Root Node is the first node in the graph, and it represents the complete sample. It can be further divided into nodes.

The features of a data collection are represented by the interior nodes, while the decision rules are represented by the branches. Finally, the outcome is represented by the Leaf Nodes. This algorithm is quite useful for resolving decision-making issues.

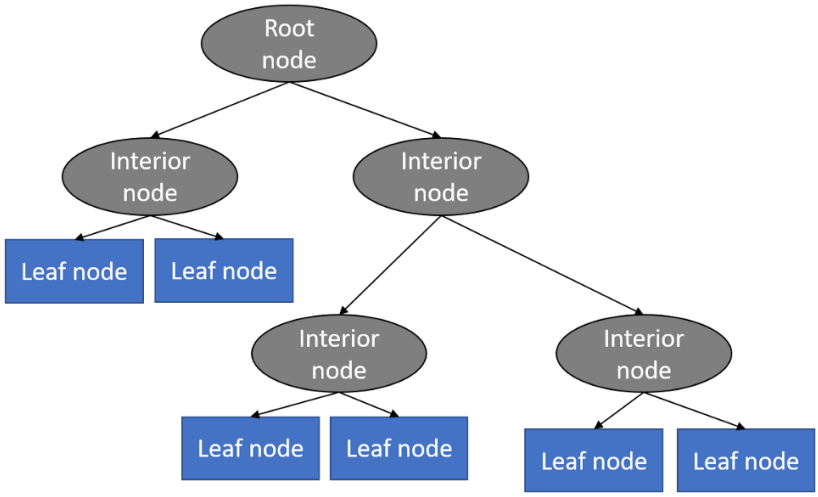


Figure 4: Decision Tree Regression

A specific data point is traversed through the entire tree by answering True/False questions until it reaches the leaf node. The average of the dependent variable's value in that particular leaf node is the final forecast. The Tree is able to estimate an appropriate value for the data point after numerous iterations.

Decision trees have an advantage that it is easy to understand, lesser data cleaning is required, non-linearity does not affect the model's performance and the number of hyper-parameters to be tuned is almost null. However, it may have an over-fitting problem, which can be resolved using the Random Forest algorithm

7.7 Feature Scaling

Feature scaling is a technique for putting the data's independent features into a set range. It is used to handle significantly changing magnitudes, values, or units during data pre-processing. If feature scaling is not done, a machine learning algorithm will assume larger values to be higher and smaller values to be lower, regardless of the unit of measurement.

So, there are two types of feature scaling and they are:-

7.7.1 Normalisation

Normalization is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling.

Here's the formula for normalization:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$

1. When the value of X is the minimum value in the column, the numerator will be 0, and hence X' is 0
2. On the other hand, when the value of X is the maximum value in the column, the numerator is equal to the denominator and thus the value of X' is 1
3. If the value of X is between the minimum and the maximum value, then the value of X' is between 0 and 1

7.7.2 Standardisation

Another scaling strategy is standardisation, in which the values are centred around the mean with a unit standard deviation.

As a result, the attribute's mean becomes zero, and the resulting distribution has a unit standard deviation.

Here's the formula for normalization:

$$X' = \frac{X - \mu}{\sigma}$$

where σ is the standard deviation of the feature values and μ is mean of the feature values.
In my project, I have used **Standardisation** for the feature scaling of the variables in the project.

7.8 Coefficient of Determination R^2 Score

The R2 score, also known as the coefficient of determination, is used to evaluate the efficacy of a linear regression model. The amount of variance in the output dependent characteristic that can be predicted based on the input independent variable (s).

It's used to examine how effectively the model reproduces observed findings, based on the ratio of total deviation of results explained by the model.

Mathematical Formula for the R^2 Score is as follows :-

$$R^2 = 1 - \frac{SS_{res}}{SS_{total}}$$

7.8.1 Interpretation

In my project for the Linear Regression R^2 Score is = 0.67 It can be referred that 67% of the changeability of the dependent output attribute can be explained by the model while the remaining 32% of the variability is still unaccounted for. R2 indicates the proportion of data points which lie within the line created by the regression equation. A higher value of R2 is desirable as it indicates better results.

7.8.2 Inferences

1. The best possible score is 1 which is obtained when the predicted values are the same as the actual values.
2. R^2 score of baseline model is 0.

3. During the worse cases, R^2 score can even be negative.

8 Codeflow and System Requirements

8.1 Codeflow

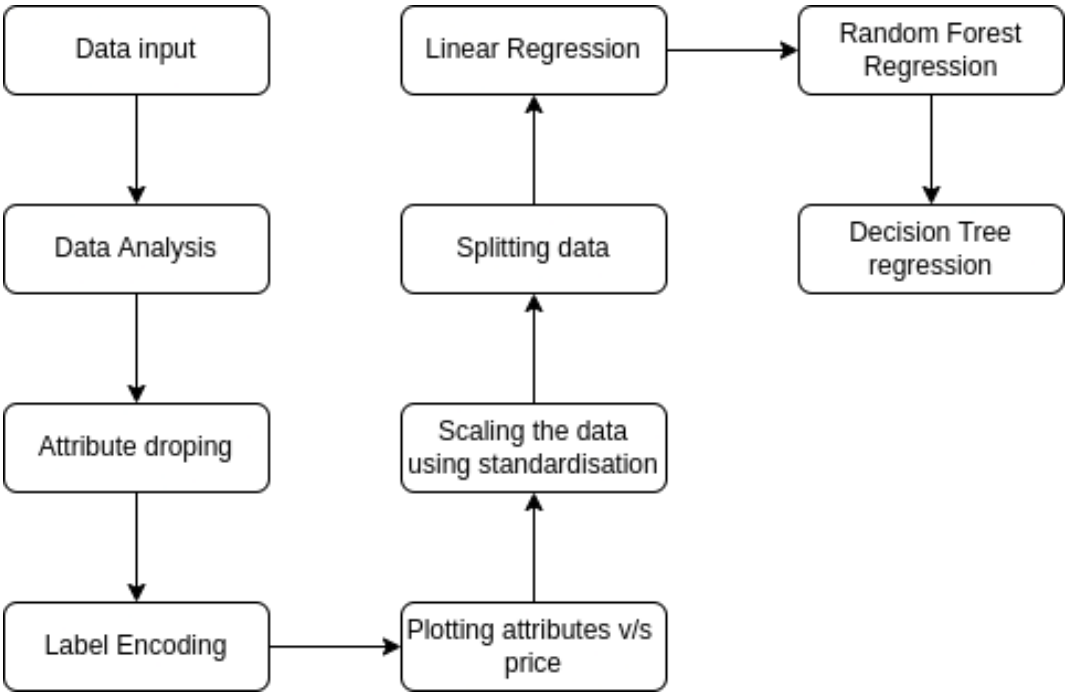


Figure 5: Code flow

8.2 System Requirements

1. Jupyter Notebooks
2. Python 3.9.7
3. Libraries
   (a) Pandas
   (b) Numpy
   (c) Sklearn
   (d) Matplotlib
   (e) Seaborn

9 Algorithms and Comparison

In the analysis, I have applied three different algorithms i.e.

1. Linear Regression

2. Decision Tree Regression

3. Random Forest Regresssion

9.1 Comparison between Linear Regression and Decision Tree Regression

1. Decision trees supports non linearity, where LR supports only linear solutions. When there are large number of features with less data-sets(with low noise), linear regressions may outperform Decision trees/random forests.

2. In general cases, Decision trees will be having better average accuracy.

3. For categorical independent variables, decision trees are better than linear regression.

4. Decision trees handles colinearity better than LR.

So from this comparison, we can inference that Decision tree regression is better than Linear Regression when there is a high noise in the data-set and also from the dataset we can see that my dataset contain some noise, some categorical values, hence Decision tree our performs the dataset.

9.2 Comparison between Random Forest Regression and Decision Tree Regression

A decision tree is a supervised machine learning algorithm that can be used for both classification and regression problems. A decision tree is simply a series of sequential decisions made to reach a specific result.The decision tree algorithm is quite easy to understand and interpret. But often, a single tree is not sufficient for producing effective results.

Random Forest is a tree-based machine learning algorithm that leverages the power of multiple decision trees for making decisions. As the name suggests, it is a "forest" of trees!

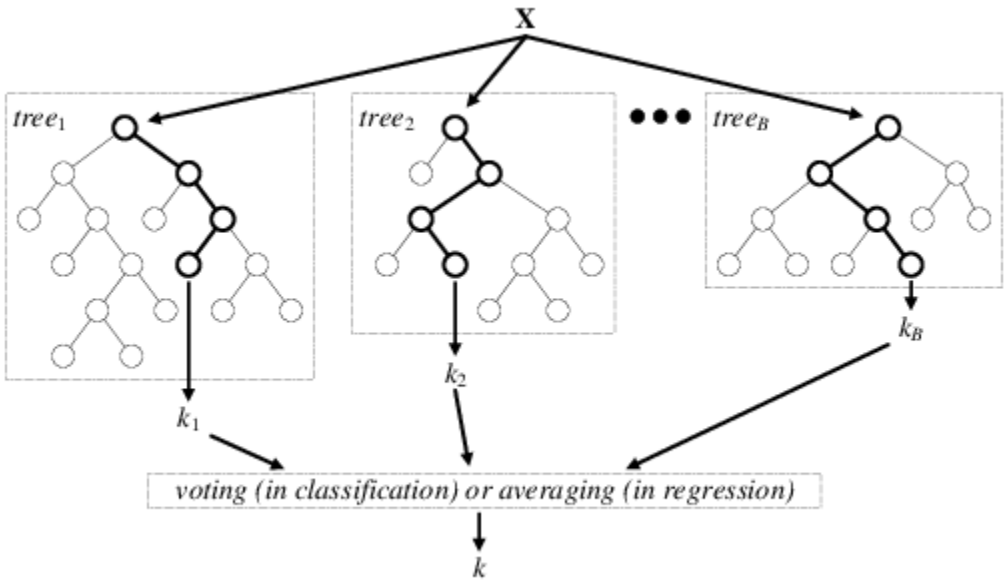


Figure 6: Random Forest

Now we can see that alone one decision tree is not useful sometimes so we can move to the Random forest Regression so that we can combine the results of the all good prediction Decision tree and hence can find the best model accuracy or we can say best accuracy.

10 Observation and Result

After applying the 3 regression algorithms, we come up with the R_2 score:-

Serial No.	Algorithm	R2 Score
1	Linear Regression	0.6702766155287029
2	Decision Tree Regression	0.8756696415771439
3	Random Forest Regression	0.9384875416895285

Table 2: Accuracy List

As we have discussed in the comparison section that Random Forest Section outperforms both Decision Tree Regression. Reason behind that is Random Forest is a tree-based machine learning algorithm that leverages the power of multiple decision trees for making decisions. As the name suggests, it is a "forest" of trees.

And from accuracy, we can see that practical inferences and theoretical inferences goes hand hand what better than that.

11 Conclusion

With the mean square error and R_2 score analysis, we see that Random Forest Regression outperforms other algorithms used with R_2 score of 0.9384875416895285 and reason is stated above in the comparison section. Dropping unnecessary attributes like CariD, manufacturers for the better model prediction.

Since the data is noisy like containing the categorical data and non scaled data, so data preprocessing techniques like one hot encoding, label encoding, data scaling using standardisation plays a important role in improving the prediction.

12 References

1. Linear Regression
2. Feature Importance
3. Decision Tree
4. Decision Tree v/s Random Forest Regression
5. Random Forest Regression
6. Decision Tree Regression
7. R_2 Score

8. Gradient Descent Algorithm
9. Feature Scaling
10. Used Car Dataset