Machine Learning Algorithms for Used Car Price Prediction System

Nirav Bhatt, Suravaram Rishabh Reddy, B Jayasurya, Prasanna Kumar, Adarsh Kumar Dubey

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India**, 600127**

niravsbhatt2002@gmail.com, rishabhsuravaram161@gmail.com, b.jayasurya123@gmail.com, kumar22maran@gmail.com, adarshdubey1357@gmail.com

ABSTRACT.

The process of selling a car at the right price is tedious and takes a toll on the general public. No one wants to sell their vehicle below its deserved price, and there is currently no model that helps the public know the selling price of their vehicles. We developed a model that will give customers the right price for their vehicles. The customer has to enter the car's specifications, such as the sunroof system, 4-wheel drive, auto parking, and various other features. The customer can also decide how recent or old the car is. Now, with all these inputs, our system will help the customer by predicting the best possible price for their vehicle by implementing machine learning. With various machine learning techniques, machine learning assists in understanding patterns in processed data. In this paper, we estimate the error based on the R squared score and root mean square error in comparison to 10 different supervised machine learning algorithm models. The results of the Random Forest Regression models show one of the least root mean squared error of 0.270555 and a high-performance R-squared value of 0.927114. Using our system, the customer needn't worry about the price of their vehicle. The customer will be able to know whether he is getting a good price for his or her vehicle or not, which will influence the customer's decision on selling the vehicle.

Keywords: Linear Regression, Ridge Regression, Lasso Regression, Bayesian Regression, Forest Tree Regression, Decision Tree Regression, Random Forest Regression, Gradient Boost Model, Light Gradient Boost Model (LGB), Extra Gradient Boost Model (XGB), Artificial Neural Networks, Multilayer Perceptron (MLP), Information Gain, Entropy, Gini index

1. Introduction

It is surely no easy task when it comes to selling your vehicle, be it as simple as two-wheelers such as bikes or scooters, or even eight-wheelers such as trucks, and the systems that exist for predicting the accurate price of a vehicle do not seem to make things any easier. It is a never-ending struggle that the general public goes through while selling their vehicle. No doubt, finding the right person to sell your vehicle is tough, but selling it at the right price is an even tougher task. Customers always think that their vehicle is the best and will end up asking a good price for it, which might be more than what the client offers to purchase the car. According to market research, the price of your vehicle will depreciate by 5% of the purchase price and may even depreciate by 10% later on, depending on a variety of factors. It's also said that the price of your car can go down to 50% of its original cost within 4 to 5 years. While all of these facts are correct, getting a good price for the customer becomes an even more difficult task. Most of the customers will sell their vehicle for a lower price than what it is capable of being sold for. The major reason is that the customer has no idea how to get information about the accuracy of the price offered, i.e., in other words, the customer will not know if the price offered is the right price or not. Hence, this led us to develop the used car price prediction model. This model will not only predict the accuracy of the price offered, but will also help the customers make a decision quicker and easier when it comes to selling their vehicle.

2. Literature Survey

Throughout the course of the project, we have referred to research papers written by various authors. Below is the list of all the research papers which have helped us to complete our project.

Sl No.	Title	Author / Journal name / Year	Technique	Result
1.	Used Cars Price Prediction using Supervised Learning Techniques	Pattabiraman Venkatasubbu, Mukkesh Ganesh /International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-9 Issue-1S3 / December 2019	Aim is to create a model, which will be able to predict price of a used car depending upon consumer data as well as various other features	The paper took the rate of prediction of different models, and was found to be less than 5 %. To get more accuracy, advanced Machine Learning models can be used.
2.	Price Prediction of Used Cars Using	Mr.Ram Prashath R, NIthish C N, Ajith Kumar J/International Journal for Research	Develop a unique model which anticipates the price of used cars based on a number of	A large number of characteristics or factors must be taken into account for accurate prediction

	Machine Learning	in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653 / May 2022	factors like vehicle model, year of manufacture, fuel type, Price, Kms Driven	which makes price prediction of the used cars a tedious task.
3.	Used Car Price Prediction Model	Praful Rane, Deep Pandya, Dhawal Kotak /International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 / Apr 2021	The system uses Regression Algorithms to provide a continuous value as the output and not as categorized value because of which predicting the actual price of the car becomes easier.	The system uses different ML algorithms to predict the accurate price of used cars.
4.	Used Car Price Prediction using Different Machine Learning Algorithms	Prof. Pallavi Bharambe, Bhargav Bagul, Shreyas Dandekar, Prerna Ingle4 / International Journal for Research in Applied Science & Engineering Technology (IJRASET) / Apr 2022	The price of used cars were determined by features such as Color of exterior, transmission type, door number, air conditioning etc.	The multiple attributes of car which is taken into account while predicting the price of cars makes the process a tedious one.
5.	Car price prediction using machine learning techniques	Enis Genic, Dino Keco, Zerina Masetic / Research Gate / February 2019	A model is built using various ML algorithms such as Random forest, SVM etc. which is used to predict the car price in countries such as Bosnia and Herzegovina.	The model was unified into Java application, which was then estimated with the help of test data giving a good accuracy of 87.38%.

6.	Prediction of Drag Force on Vehicles in a Platoon Configuration Using Machine Learning	Farwa Jaffar, Taha Farid, Muhammad Sajid, Yasar Ayaz, and Muhammad Jawad Khan / IEEE / November 17, 2020	Here, price of the vehicles are predicted using many ML algorithms which are applied in areas such as vehicle platooning.	In this paper, a well and through estimation of various aerodynamic coefficients for numerous vehicle platoon configurations was performed.
7.	Predicting the Price of Used Cars using Machine Learning Technique	Pudaruth Sameerchand (2014)/ International Journal of Information & Computation Technology. 4. 753-764.	Accurate price of cars used in Mauritius are predicted by looking through the various Machine Learning models.	Different algorithms are taken and various predictions are made with the help of them. In the end, it's found out which ones are accurate.
8.	Used Car Price Prediction	Abhishek Jha, Dr. Ramveer Singh, Manish, Imran Saifi, Shipra Srivastava. Used car price prediction, International Journal of Advance Research, Ideas and Innovations in Technology, www.IJARIIT.com.	Dataset has been taken from cardekho.com and various ML algorithms are used for predicting price of a car with the help of Python, flask, HTML etc.	The model is able to predict accuracy of the cars even though the car market is highly volatile.
9.	Price Prediction of Used Cars Using Machine Learning	Chuyang Jin/2021 IEEE International Conference on Emergency Science and Information Technology	Here, A model is built for the prediction of car price based on multiple aspects such as The Year of manufacturing of the vehicle, mileage given by the vehicle,	The system which is proposed, will find out the price of different sets of vehicles by analyzing multiple factors. Accuracy was found out to be 95%.

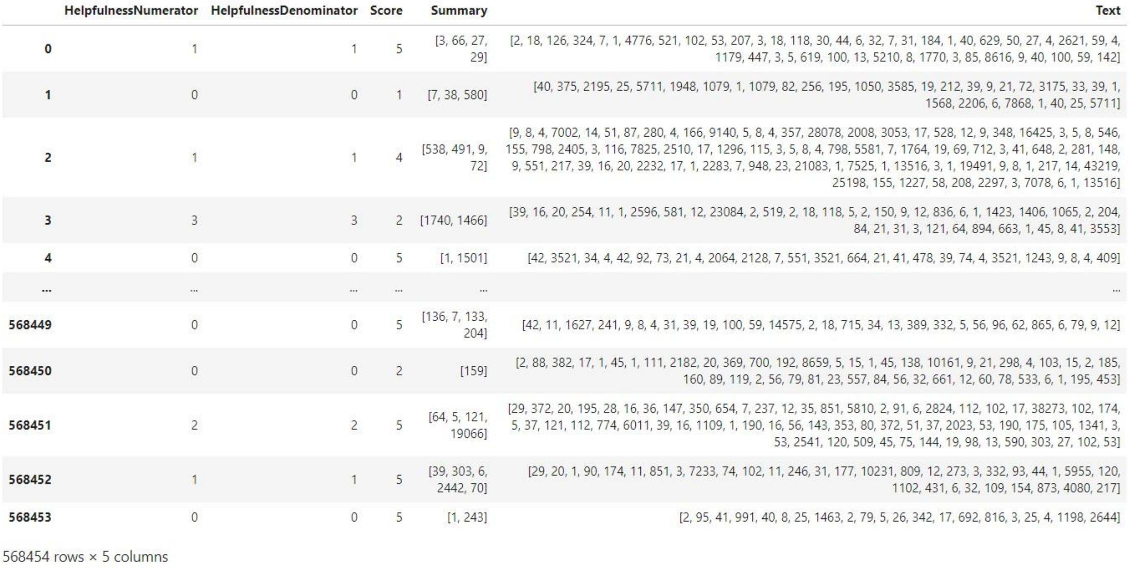
		(ICESIT)	consumption of fuel among others.	

3. Dataset Description:

The vehicle data set is obtained from Kaggle with 568454 rows and 10 columns. The data has been collected from Amazon, which focuses solely on the product description, summary, score and helpfulness. The majority of the pertinent information that Amazon gives about product description is included in this data.

4. Data Pre-Processing:

The understanding of the data and the preparation of the data are crucial steps in constructing a model because they reveal what adjustments or alterations need to be made before designing and implementing the model. Figure.4 shows the original dataset.



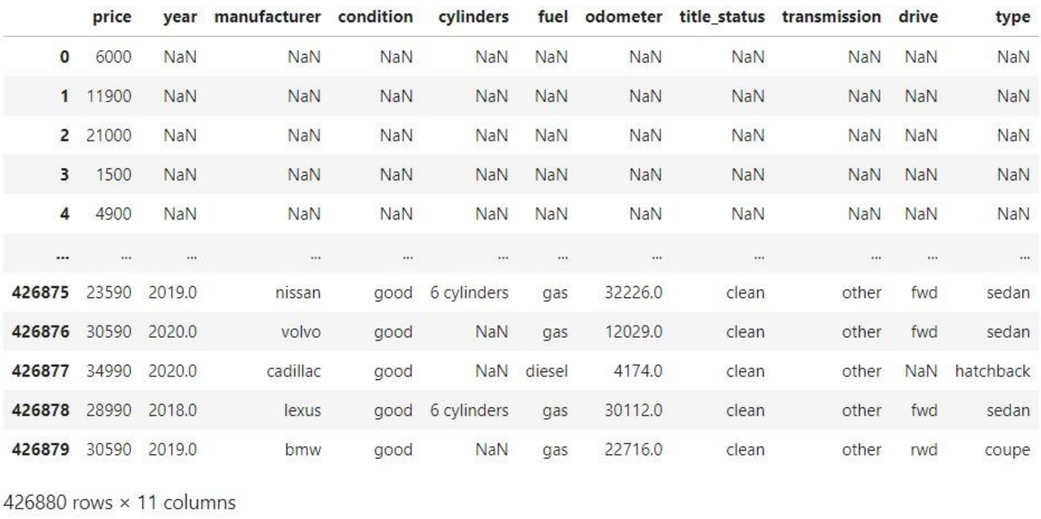
	HelpfulnessNumerator	HelpfulnessDenominator	Score	Summary	Text
0	1	1	5	[3, 66, 27, 29]	[2, 18, 126, 324, 7, 1, 4776, 521, 102, 53, 207, 3, 18, 118, 30, 44, 6, 32, 7, 31, 184, 1, 40, 629, 50, 27, 4, 2621, 59, 4, 1179, 447, 3, 5, 619, 100, 13, 5210, 8, 1770, 3, 85, 8616, 9, 40, 100, 59, 142]
1	0	0	1	[7, 38, 580]	[40, 375, 2195, 25, 5711, 1948, 1079, 1, 1079, 82, 256, 195, 1050, 3585, 19, 212, 39, 9, 21, 72, 3175, 33, 39, 1, 1568, 2206, 6, 7868, 1, 40, 25, 5711]
2	1	1	4	[538, 491, 9, 72]	[9, 8, 4, 7002, 14, 51, 87, 280, 4, 166, 9140, 5, 8, 4, 357, 28078, 2008, 3053, 17, 528, 12, 9, 348, 16425, 3, 5, 8, 546, 155, 798, 2405, 3, 116, 7825, 2510, 17, 1296, 115, 3, 5, 8, 4, 798, 5581, 7, 1764, 19, 69, 712, 3, 41, 648, 2, 281, 148, 9, 551, 217, 39, 16, 20, 2232, 17, 1, 2283, 7, 948, 23, 21083, 1, 7525, 1, 13516, 3, 1, 19491, 9, 8, 1, 217, 14, 43219, 25198, 155, 1227, 58, 208, 2297, 3, 7078, 6, 1, 13516]
3	3	3	2	[1740, 1466]	[39, 16, 20, 254, 11, 1, 2596, 581, 12, 23084, 2, 519, 2, 18, 118, 5, 2, 150, 9, 12, 836, 6, 1, 1423, 1406, 1065, 2, 204, 84, 21, 31, 3, 121, 64, 894, 663, 1, 45, 8, 41, 3553]
4	0	0	5	[1, 1501]	[42, 3521, 34, 4, 42, 92, 73, 21, 4, 2064, 2128, 7, 551, 3521, 664, 21, 41, 478, 39, 74, 4, 3521, 1243, 9, 8, 4, 409]
...	...	...	...	...	...
568449	0	0	5	[136, 7, 133, 204]	[42, 11, 1627, 241, 9, 8, 4, 31, 39, 19, 100, 59, 14575, 2, 18, 715, 34, 13, 389, 332, 5, 56, 96, 62, 865, 6, 79, 9, 12]
568450	0	0	2	[159]	[2, 88, 382, 17, 1, 45, 1, 111, 2182, 20, 369, 700, 192, 8659, 5, 15, 1, 45, 138, 10161, 9, 21, 298, 4, 103, 15, 2, 185, 160, 89, 119, 2, 56, 79, 81, 23, 557, 84, 56, 32, 661, 12, 60, 78, 533, 6, 1, 195, 453]
568451	2	2	5	[64, 5, 121, 19066]	[29, 372, 20, 195, 28, 16, 36, 147, 350, 654, 7, 237, 12, 35, 851, 5810, 2, 91, 6, 2824, 112, 102, 17, 38273, 102, 174, 5, 37, 121, 112, 774, 6011, 39, 16, 1109, 1, 190, 16, 56, 143, 353, 80, 372, 51, 37, 2023, 53, 190, 175, 105, 1341, 3, 53, 2541, 120, 509, 45, 75, 144, 19, 98, 13, 590, 303, 27, 102, 53]
568452	1	1	5	[39, 303, 6, 2442, 70]	[29, 20, 1, 90, 174, 11, 851, 3, 7233, 74, 102, 11, 246, 31, 177, 10231, 809, 12, 273, 3, 332, 93, 44, 1, 5955, 120, 1102, 431, 6, 32, 109, 154, 873, 4080, 217]
568453	0	0	5	[1, 243]	[2, 95, 41, 991, 40, 8, 25, 1463, 2, 79, 5, 26, 342, 17, 692, 816, 3, 25, 4, 1198, 2644]

568454 rows × 5 columns

Fig. 1: The Representation of the original dataset

It’s said that Data Scientists spend almost around 80% of the total project allocated time only for Pre-Processing. This step is the most important step when it comes to building a Machine Learning model. It is always important to remember that the better the input given to the model, the better the output, and hence better the accuracy of the model. Datasets sometimes lack particular information on activity or trends and also contain erroneous information. As result, this can result in faulty data gathering, which could then lead to poor models that are built using the data. The data can be pre-processed to address such issues. We can either go for modification of the various inputs which have been provided in the dataset or go for encoding. This process is part of the Pre-Processing phase, and it will help the data easier to parse for the computer. Hence, the Machine Learning models we choose to apply will be able to easily understand as well as comprehend the data, which will help us to provide better results. In this project, the following steps are performed to preprocess the dataset:

The first stage is to eliminate aspects that are irrelevant or worthless attributes, such as posting_date, county, description, image_url, VIN, url, region_url, id, model, state, lat, long, region, paint_color and size from the dataset. Fig.5. shows the dataset after dropping columns.



	price	year	manufacturer	condition	cylinders	fuel	odometer	title_status	transmission	drive	type
0	6000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	11900	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	21000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	1500	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	4900	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...
426875	23590	2019.0	nissan	good	6 cylinders	gas	32226.0	clean	other	fwd	sedan
426876	30590	2020.0	volvo	good	NaN	gas	12029.0	clean	other	fwd	sedan
426877	34990	2020.0	cadillac	good	NaN	diesel	4174.0	clean	other	NaN	hatchback
426878	28990	2018.0	lexus	good	6 cylinders	gas	30112.0	clean	other	fwd	sedan
426879	30590	2019.0	bmw	good	NaN	gas	22716.0	clean	other	rwd	coupe

426880 rows × 11 columns

Fig. 2: Representation of the Dataset after dropping the columns

As a next step, check for missing values for each feature. The below figure, Fig.3. shows the missing values in the dataset.

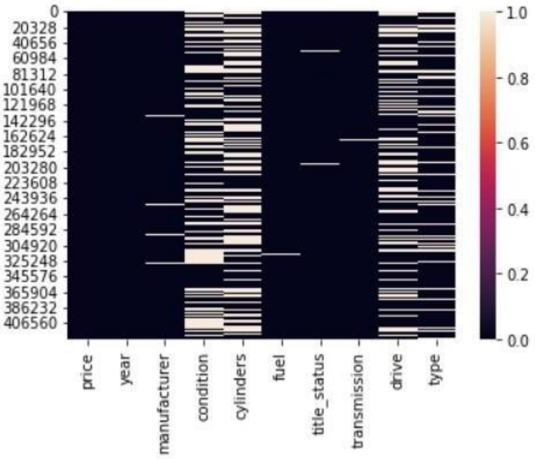
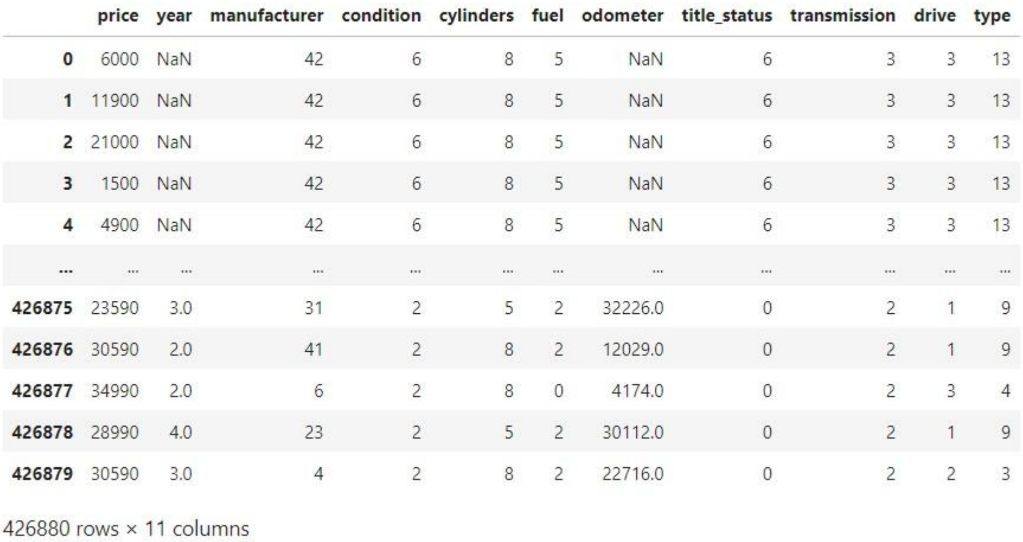


Fig. 3: Representation of Missing values in the dataset

Label Encoder:

Now, after the Pre-Processing steps, our dataset contains 12 characteristic categorical columns with 4 numerical columns. We cannot apply different models of Machine Learning to the categorical columns, and hence, we will have to convert these categorical variables into numerical variables. The LabelEncoder software from the Sklearn library will be used to solve the above pressing issue. The below figure, Fig.4 depicts label encoding, which is performed on the dataset.



	price	year	manufacturer	condition	cylinders	fuel	odometer	title_status	transmission	drive	type
0	6000	NaN	42	6	8	5	NaN	6	3	3	13
1	11900	NaN	42	6	8	5	NaN	6	3	3	13
2	21000	NaN	42	6	8	5	NaN	6	3	3	13
3	1500	NaN	42	6	8	5	NaN	6	3	3	13
4	4900	NaN	42	6	8	5	NaN	6	3	3	13
...	...	...	...	...	...	...	...	...	...	...	...
426875	23590	3.0	31	2	5	2	32226.0	0	2	1	9
426876	30590	2.0	41	2	8	2	12029.0	0	2	1	9
426877	34990	2.0	6	2	8	0	4174.0	0	2	3	4
426878	28990	4.0	23	2	5	2	30112.0	0	2	1	9
426879	30590	3.0	4	2	8	2	22716.0	0	2	2	3

426880 rows × 11 columns

Fig. 4: The Representation of the dataset after Label Encoding

Below Figure, Fig. 5 depicts the heat map which shows the correlation of the dataset.

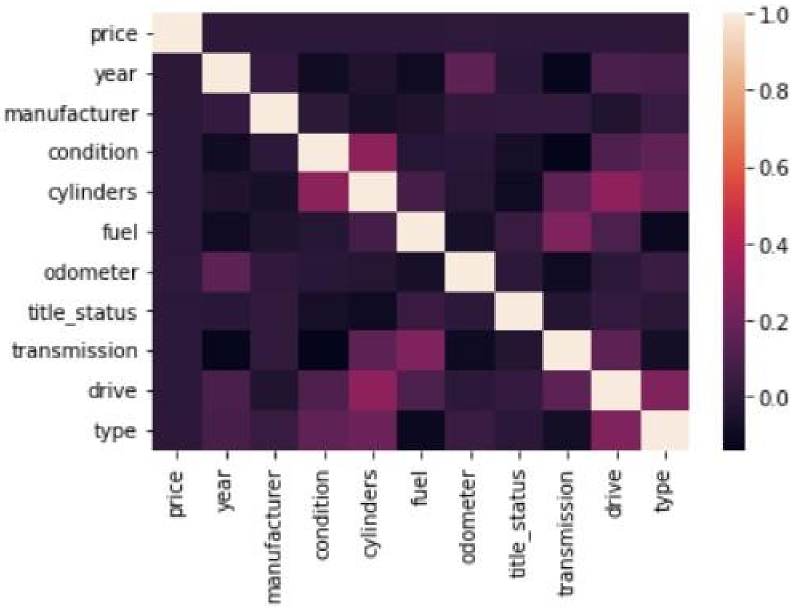


Fig. 5: The Heat Map depicting the correlation of the dataset

To simplify the year attribute, subtract the year from the current year, which is 2022. The Iterative Imputer method with ascending imputation order and random state 1 is used to impute missing values, and different estimators are applied. The MSE of each estimator is calculated using cross_val_score.

The difference between the third and first quartiles is used to discover the upper and lower limits of the property.

Outlier is usually a value, or we could say an observation that is separated from the bunch of values obtained from a set of datapoints. Outliers can occur mainly due to the measurement errors, or due to faulty inputs given to the model. They can occur due to processing errors as well as poor sampling in the dataset. Outliers are removed in both the price and odometer attributes. The below group of equations represent the equations of the different attributes of the Box plot.

It is important to remember that IQR stands for Inter-Quartile Range, whereas Q1, Q2, and Q3 stand for the first Quartile, second Quartile and the third Quartile respectively.

$$IQR = Q3 - Q1 \ldots\ldots\ldots(1)$$

Eq. 1: Equation representing Box Plot’s Inter-Quartile Distance

$$Lower_{limit} = Q1 - 1.5IQR \ldots\ldots\ldots(2)$$

Eq. 2: Equation representing Box Plot’s Lower Limit

$$Upper_{limit} = Q3 + 1.5IQR\text{.........}(3)$$

Eq. 3: Equation representing Box plot's Upper Limit

Now, to understand the definition of Outliers better, we could say in simpler words that, Outlier is any value or observation that is lesser than the $Lower_{limit}$ or greater than $Upper_{limit}$.

5. Methodology

The model algorithm used in this paper is from Scikit-Learn, which is a library present in Python used mainly for Machine Learning. Scikit-learn has a lot of features such as Classification, Regression and even Clustering Algorithms. The model is created using the training set. The training set should not be overfitted in machine learning algorithms since it will increase the errors in predicting values or give biased output.

In our paper, we have used supervised Machine Learning algorithms such as Multiple Linear regression, Bayesian regression, Ridge regression, and Lasso regression.

The supervised machine learning models used in this paper are Multiple linear regression, Bayesian regression, Ridge Regression, and Lasso Regression. The tree models used in this are Decision tree regression and the ensemble models used in this paper are Light Gradient Boosting Machine (LGBM), Extreme Gradient Boosting, Random Forest regression, and Gradient boosted regression. The Artificial Neural Network (ANN) models used in this paper are Multi-layer Perceptron regressor.

Regression Models used in our project:

A. Multiple Linear Regression:

The Multiple Linear regression model is the simple, quickest and easiest regression model, but its precision is lower than other models and it is used in certain problems. It uses weights to fit the training data in a straight line, and the weights are adjusted by minimizing the mean squared error loss.

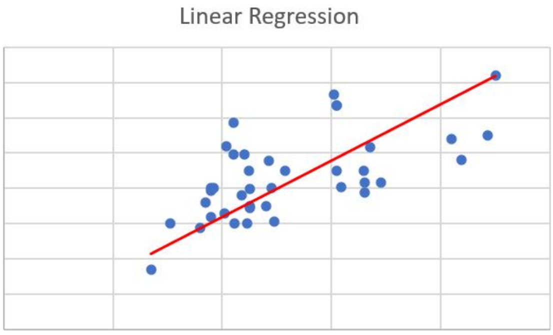


Fig. 6. Simple graph depicting Linear Regression

The formula for Multiple Linear Regression is given below in the Equation 4.

$$y_i = b_0 + b_1 x_1 + b_2 x_2 + \dots + b_n x_n + \epsilon \text{ ....(4)}$$

Eq. 4: Equation representing Multiple Linear Regression

where, y_i is the target variable, or dependent variable, and x_i are the predictors, or independent variables. b_0 is the y-intercept, b_i is the slope coefficient for the respective independent variables, and ϵ is the error term.

In case of two independent variables, the formula for the slope coefficients will be as shown below.

$$\widehat{b1} = \frac{\left(\sum x_2^2\right) * \left(\sum x_1 y\right) - \left(\sum x_1 x_2\right)\left(\sum x_2 y\right)}{\left(\sum x_1^2\right)\left(\sum x_2^2\right) - \left(\sum x_1 x_2\right)^2} \text{ ....(5)}$$

Eq. 5: Equation depicting the slope coefficient for the first independent variable

$$\widehat{b2} = \frac{\left(\sum x_1^2\right) * \left(\sum x_2 y\right) - \left(\sum x_1 x_2\right)\left(\sum x_1 y\right)}{\left(\sum x_1^2\right)\left(\sum x_2^2\right) - \left(\sum x_1 x_2\right)^2} \text{.........(6)}$$

Eq. 6: Equation depicting the slope coefficient for the second independent variable

B. Ridge Regression:

Ridge Regression is a regularized version of the linear regressor, but it has an alpha parameter to control the regulation of the model and reduce the variance of the estimates. It is used to tune for multiple correlation datasets with two or more input attributes. Ridge regression performs L2 regularization. The higher the alpha value, the more the coefficients are shrunk to reduce the complexity. Equations 7 and Equation 8 below shows the formula for applying Ridge Regression.

$$Cost(W) = RSS(W) + \lambda * (sum\ of\ Squares\ of\ Weight)\text{.........}(7)$$

Eq. 7: Equation representing Ridge Regression

$$\sum_{i=1}^{N}\left\{y_i - \sum_{j=0}^{M} w_j\ * x_{ij})\right\}^2 + \lambda\left(\sum_{j=0}^{M} w^2{}_j\right)\text{.........}(8)$$

Eq. 8: Alternative Equation representing Ridge Regression

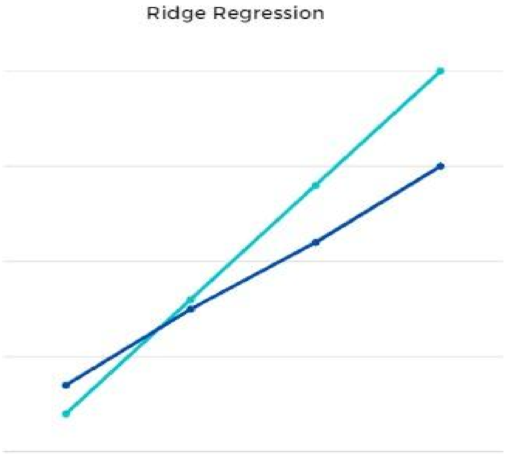


Fig. 7. Simple graph depicting Ridge Regression

C. Lasso Regression:

LASSO is Least Absolute Shrinkage and Selection Operator is a shrinkage regularization technique to avoid overfitting them. It is a simple sparse model, which has fewer parameters than other regressions. This type of regression is used when the dataset shows high multicollinearity or when you want to automate variable elimination and feature selection. Equation 9 below shows the formula for applying Lasso Regression.

$$\sum_{i=1}^{M}(y_i - y_i^{'})^2 = \sum_{i=1}^{M}(y_i - \sum_{j=0}^{n}\beta_j * x_{ij})^2 + \lambda * \left(\sum_{j=0}^{n}|\beta_j|\right) \text{.........(9)}$$

Eq. 9: Equation depicting Lasso Regression

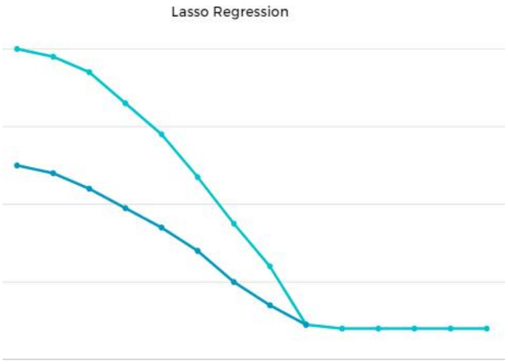


Fig. 8: Simple graph depicting Lasso Regression

D. Bayesian Ridge:

Bayesian regression is a linear regression that can be implemented with insufficient or poorly distributed data using probability distributions rather than point estimates.

Tree Models:

A. Decision Tree Forest:

The decision tree divides the data set into subgroups based on the attribute value test, and then the procedure is iterated recursively. It is a model that does not require any prior domain knowledge or parameter settings to be generated. It is a supervised learning technique that can handle high-dimensional data accurately.

Decision Trees come in handy when we are handling missing values in huge sets of data. It is more effective and provides optimum accuracy with a minimum number of features. It is capable of handling both continuous and categorical datasets and does not require data preprocessing so it easy to use. The below figure, Fig. 4 depicts the simplest form of a Decision Tree

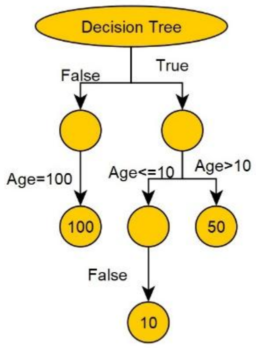


Fig. 9: Figure Depicting a Sample Decision Tree

The information gain function is used to measure the reduction of entropy. The below equations, Eq. 10 and Eq. 11 represent Information Gain and Entropy respectively.

$$\text{Information Gain} = \text{Entropy}(\text{S}) - [(\text{Weighted Avg}) * \text{Entropy}(\text{each feature}) \ldots\ldots\ldots(10)$$

Eq. 10: Equation depicting Information Gain

$$\text{Entropy}(\text{s}) = -\text{P}(\text{yes})\text{log2 P}(\text{yes}) - \text{P}(\text{no})\text{log2 P}(\text{no}) \ldots\ldots\ldots(11)$$

Eq. 11: Equation representing Entropy

The Gini Index will measure the purity or the impurity of the node, which will help in the determining till what extent the output is misclassified randomly. The Equation below, Eq. 12 represents the Gini Index Formula.

$$Gini = 1 - \sum_{i=0}^{c-1} [p_t]^2 \ldots\ldots\ldots(12)$$

Eq. 12: Equation representing Gini Index

B. Random Forest Regression:

Random forest regression is a classifier based on a collection or ensemble of decision tree model that is used in the decision tree framework to create a random decision tree. The model is developed using bagging by combining multiple decision trees, which estimates the value in each decision tree, and the output is determined by the majority vote on the outcome. The random tree accuracy is comparable to the single decision tree accuracy, but the model in the random forest tree will not be overfitted. It has better prediction accuracy than normal regression and more powerful. The below Figure, Fig. 10 represents a collection of multiple decision trees, or in other words a simple representation of Random Forest.

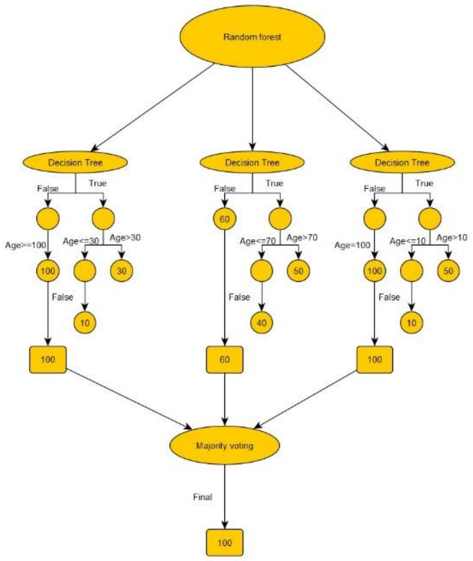


Fig. 10: Figure representing a simple representation of Random Forest

C. Gradient Boosted Regression:

Gradient boosted regression is an ensemble model, similar to random forest regression, but gradient boosted creates multiple trees to improve the deficiencies in the previous tree, resulting in the lowest absolute mean error. The learning rate and loss function can be changed. The learning rate should not be high, so the tree will not be overfitted. The below Equation, Eq. 13 represents the Gradient Boosted Regression.

$$F_0(x) = \text{arg}_\gamma \, min \sum_{i=1}^{n} L(y, \gamma) \text{ ....(13)}$$

Eq. 13: Equation depicting Gradient Boosted Regression

We have another equation here, Eq. 14 which depicts Pseudo residuals:

$$r_{im} = -\left[\frac{\partial L\big(y_i, F(x_i)\big)}{\partial F(x_i)}\right] \text{.........}(14)$$

Eq. 14: Equation depicting Pseudo residuals

D. XG Boost (XG):

XG Boost ensemble learning models use a loss function and a regularization term. The Equation below, Eq. 15 will help in understanding XG Boost better.

$$Obj^{(t)} = -\frac{1}{2}\sum_{j=1}^{T}\frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \epsilon I_j} h_i + \gamma} + \gamma T \text{ ....}(15)$$

Eq. 15: Equation depicting XG Boost

We have another formula representing the Gain. Equation below, Eq. 16 will show this.

$$Gain = \frac{1}{2}\left[\frac{(\sum_{i \epsilon I_L} g_i)^2}{\sum_{i \epsilon I_L} h_i + \gamma} + \frac{(\sum_{i \epsilon I_R} g_i)^2}{\sum_{i \epsilon I_R} h_i + \gamma} + \frac{(\sum_{i \epsilon I_I} g_i)^2}{\sum_{i \epsilon I_I} h_i + \gamma}\right] - \gamma \text{.........}(16)$$

Eq. 16: Equation depicting Gain

E. Light Gradient Boosting Machine (LGBM):

Light gradient-boosting ensemble machine learning models, which have XG Boost advantages.

The below equation, Eq. 17 depicts Entropy or Gini-index.

$$\widetilde{V}_j(d) = \frac{1}{n}\left(\frac{\left(\sum_{x_i \epsilon A_l} g_i + \left(\frac{1-a}{b}\right)\sum_{x_i \epsilon B_l} g_i\right)^2}{n_l^j(d)} + \frac{\left(\sum_{x_i \epsilon A_r} g_i + \left(\frac{1-a}{b}\right)\sum_{x_i \epsilon B_r} g_i\right)^2}{n_r^j(d)}\right) \text{.........}(17)$$

Eq. 17: Equation depicting Entropy

Artificial Neural Networks (ANN):

A. Multi-layer Perceptron (MLP):

Multi-layer Perceptron (MLP) Regression algorithm comes under supervised learning that emulates a function from the input dataset. It is an artificial neural network composed of multiple layers of perceptron. This model can be used for conventional regression as well as more complicated issues such as image recognition.

Layers are classified into three types: input layers, output layers, and hidden layers. The data is received by the input layer and then transferred through one or more hidden layers, each with a variable number of neurons that calculate and alter the input data to produce meaningful output. The answer to the user's input is delivered by the output layer. It uses hidden layers and neurons to predict the value of the unknown input. It can perform not only supervised learning, but also unsupervised learning. The figure below depicts ANN.

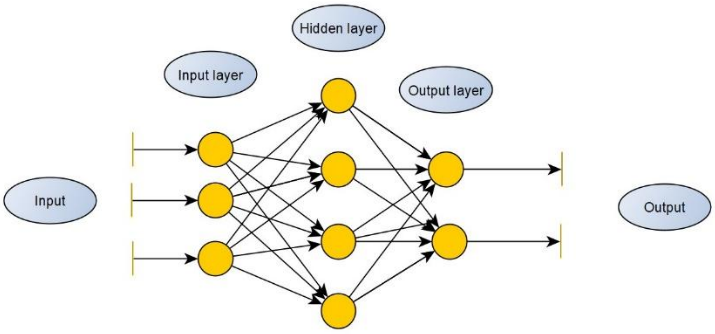


Fig. 11: A Simple figure depicting ANN

The following equation, Eq. 18 depicts Transfer function:

$$p_j(t) = \sum_i o_i\,(t) w_{ij} \text{.........}(18)$$

Eq. 18: Equation depicting Transfer Equation

The cost function can be optimized using either of the two methods mentioned below.

1) Back Propagation

Backpropagation compares the output to the desired output after assigning weight to neurons, and the models produce an output. The cost function is decreased by shifting the weights from the last layer's neurons to the first layer's neurons.

2) Forward Propagation

After assigning weight to the different neurons in the network, the model produces an output. When the output is produced, the method compares the output we have just gotten to the desired output.

6. Experimental Analysis:

A comparative study of all the prediction algorithms is done to find out which model is the best when it is used to predict using the given dataset. The performance of the regression models can be determined by various parameters, some of which are mean absolute errors, mean square errors, and the r^2 ratio.

A. Mean Squared Error (MSE):

MSE is the most widely used loss function used for regression. It is the average sum of the squared distances between the original value and the predicted value. The below equation, i.e., Equation 19 shows the formula for applying Mean Squared Error.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_j - \tilde{y_j}\right)^2 \text{.........(19)}$$

Eq. 19: Equation depicting Mean Squared Error

B. Median Absolute Deviation (MAD):

Median Absolute Error is the robust measure of error between our target values and predicted values. The below equation, i.e., equation 20 shows the formula for applying Mean Absolute Error.

$$MAD = Median(\left|y_j - \tilde{y_j}\right|) \text{.........(20)}$$

Eq. 20: Equation depicting Median Absolute Error

C. Root Mean Square (RMS):

Root Mean Square is the square root of the arithmetic mean of the individually squared terms of a set. The below equation, i.e., equation 21 shows the formula for applying Root Mean Square Error.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_j - \tilde{y_j})^2}{N}} \text{.........(21)}$$

Eq. 21: Equation depicting Root Mean Square Error

D. R Square (R^2):

R-squared is a statistical measure that shows how much of a dependent variable's variance is explained by one or more independent variables in a regression model.

$$R^2 = \frac{n\left(\sum_{i=1}^{N} x_i y_i\right) - \left(\sum_{i=1}^{N} x_i\right) * \left(\sum_{i=1}^{N} y_i\right)}{\sqrt{\left[n * \left(\sum_{i=1}^{N} x_i{}^2 - \left(\sum_{i=1}^{N} x_i\right)^2\right)\right] * \left[n * \left(\sum_{i=1}^{N} y_i{}^2 - \left(\sum_{i=1}^{N} y_i\right)^2\right)\right]}} \text{.........}(22)$$

Eq. 22: Equation depicting R Squared Error

Please note that the image on the left depicts red dots that represent the predicted value along with blue dots that represent the original value. The image on the right with blue dots shows the absolute difference between the original and predicted values.

7. Observation:

Regression Models:

A. Multiple Linear Regression:

The fit intercept parameter is configured. We obtained the R-squared score of 0.517725 and the root mean squared of 0.695954 from the model. The below graph depicts both the predicted price and the actual price obtained through Linear Regression.

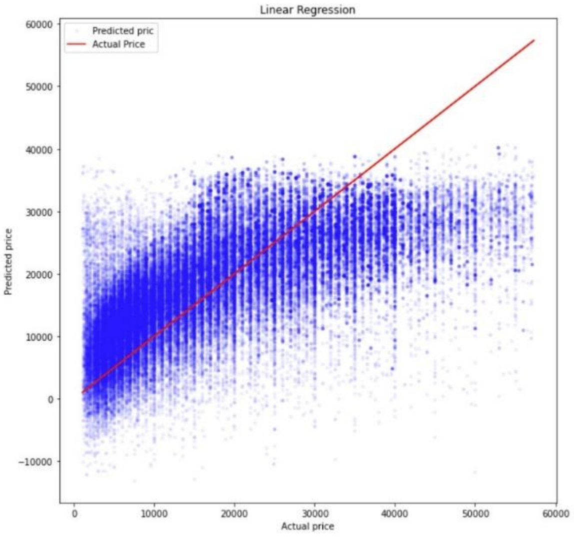


Fig. 12: Representation of the predicted price and actual price in the Linear Regression Model

B. Ridge Regression:

The fit intercept parameter is configured to random state 1. We obtained the R-squared score of 0.517725 and the root mean squared of 0.695954 from the model. The below graph depicts both the predicted price and the actual price obtained through Ridge Regression.

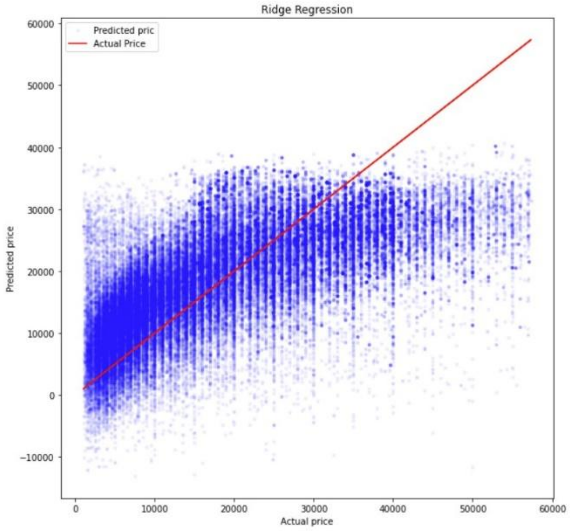


Fig. 13: Representation of the predicted price and actual price in the Ridge Regression Model

C. Lasso Regression:

The fit intercept parameter is configured to random state 1. We obtained the R-squared score of 0.517725 and the root mean squared of 0.695954 from the model. The below graph depicts both the predicted price and the actual price obtained through Lasso Regression.

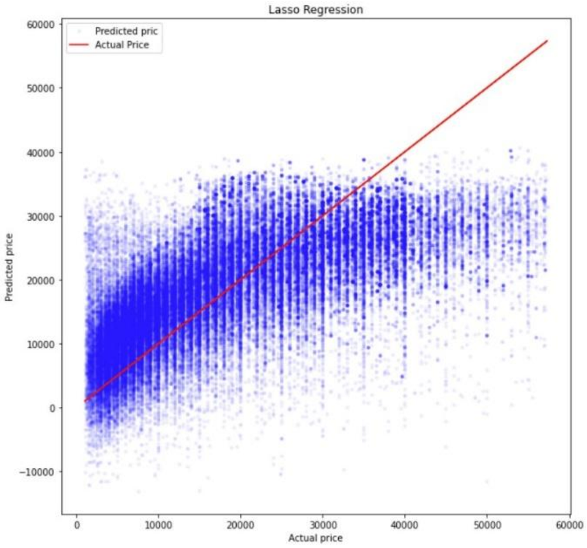


Fig. 14: Representation of the predicted price and actual price in the Lasso Regression Model

D. Bayesian Regression:

The fit intercept parameter is configured. We obtained the R-squared score of 0.517725 and the root mean squared of 0.695954 from the model. The below graph depicts both the predicted price and the actual price obtained through Bayesian Regression.

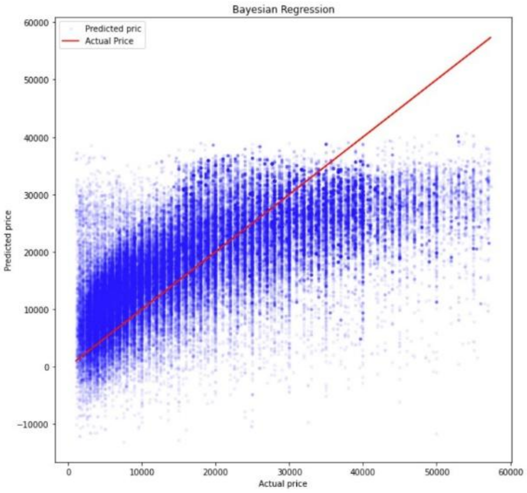


Fig. 15: Representation of the predicted price and actual price in the Bayesian Regression Model

E. Decision Tree Regressor:

The decision tree regression model is configured with the maximum feature to auto and the random state configured to 1 and by using the trial & error method to determine which depth has the least error, it is determined that depth 18 has the lowest error. We obtained the R-squared score of 0.854662 and the root mean squared of 0.382053 from the model. The below graph depicts both the predicted price and the actual price obtained through Decision Tree Regressor.

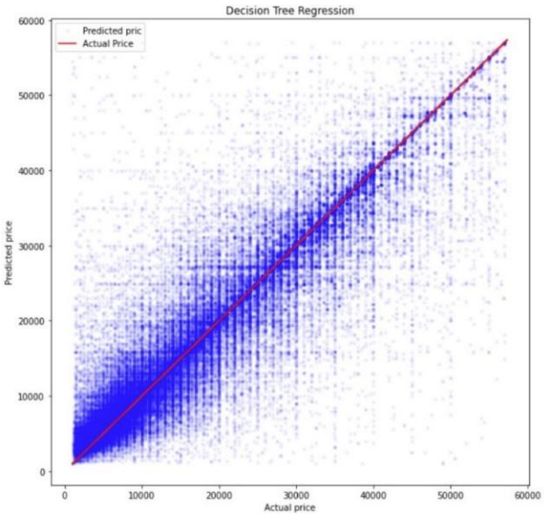


Fig. 16: Representation of the predicted price and actual price in the Decision Tree Regressor

F. Random Forest Regressor:

The random forest regression model is configured with the n estimators to 100, the max features to log2, and the random state configured to 1 and by using the trial & error method to determine which depth has the least error, it is determined that depth 30 has the lowest error. We obtained the R-squared score of 0.927114 and the root mean squared of 0.270555 from the model. The below graph depicts both the predicted price and the actual price obtained through Random Forest Regressor.

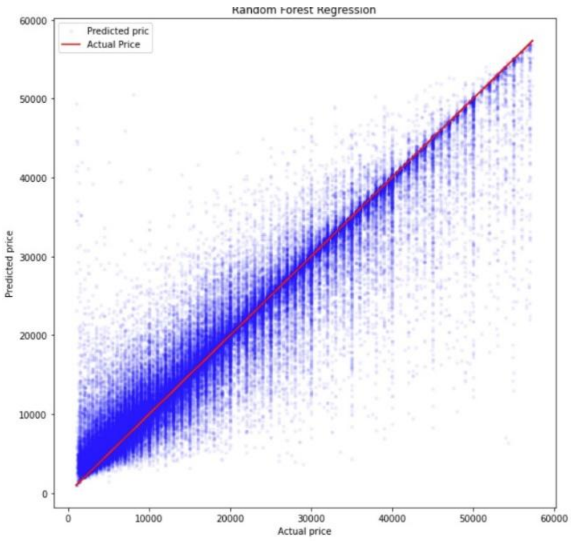


Fig. 17: Representation of the predicted price and actual price in Random Forest Generator

G. XGB Regression Model:

The XGB regression model is configured with the random state configured to 1 and by using the trial & error method to determine which depth has the least error, it is determined that depth 15 has the lowest error. We obtained the R-squared score of 0.929676 and the root mean squared of 0.265757 from the model. The below graph depicts both the predicted price and the actual price obtained through XGB Regressor.

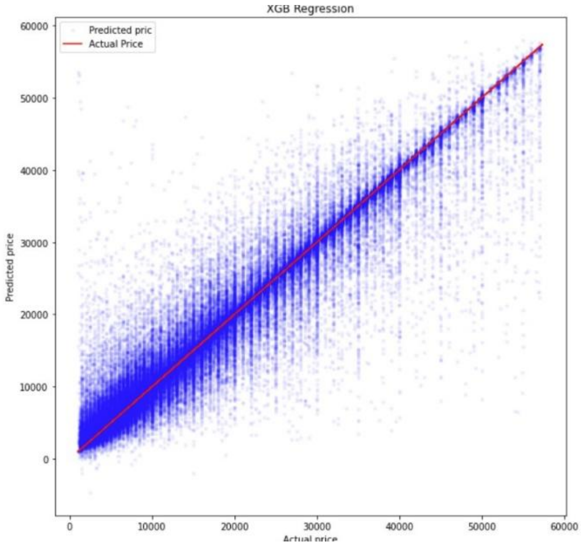


Fig. 18: Representation of the predicted price and actual price in the XGB Regression Model

H. LGBM Regression Model:

The LGB regression model is configured with the random state configured to 1 and by using the trial & error method to determine which depth has the least error, it is determined that depth 10 has the lowest error. We obtained the R-squared score of 0.834691 and the root mean squared of 0.407457 from the model. The below graph depicts both the predicted price and the actual price obtained through LGB Regressor.

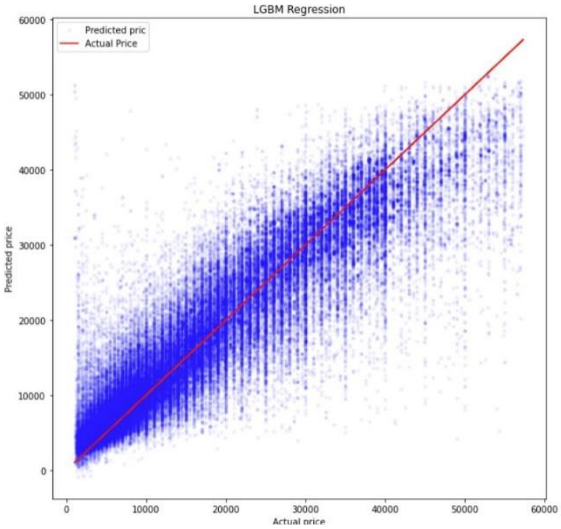


Fig. 19: Representation of the actual price and predicted price obtained through the LGBM Regression Model

I. Gradient Boosting Regressor:

The Gradient Boosting regression model is configured with the random state configured to 1 and by using the trial & error method to determine which depth has the least error, it is determined that depth 16 has the lowest error. We obtained the R-squared score of 0.92784 and the root mean squared of 0.269204 from the model. The below graph depicts both the predicted price and the actual price obtained through Gradient Boosting Regressor.

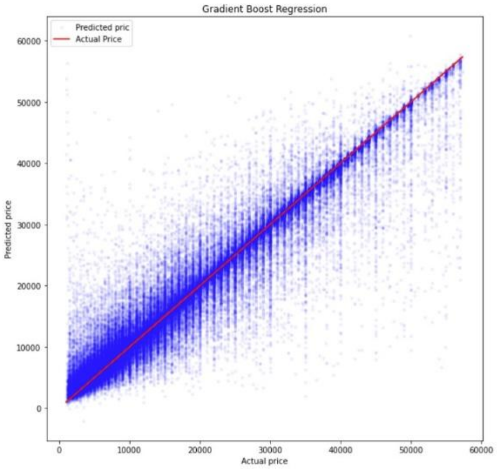


Fig. 20: Representation of the actual price and predicted price obtained through Gradient Boost Regression

J. Multi-layer Perceptron (MLP):

The MLP regression model is configured with four hidden layers of 28 neurons in layers 1, 2, and 42 neurons in layer 3, and one neuron in layer 4, with a maximum iteration of 350, the activation being a rectified linear unit function, the learning rate being adaptive, the solver being adaptive moment estimation, the random state configured to 1, and the solver iterating to a maximum of 350 where the weights of the neurons are changed in each iteration. We obtained the R-squared score of 0.751537 and the root mean squared of 0.499533 from the model. The below graph depicts both the predicted price and the actual price obtained through the MLP Regressor.

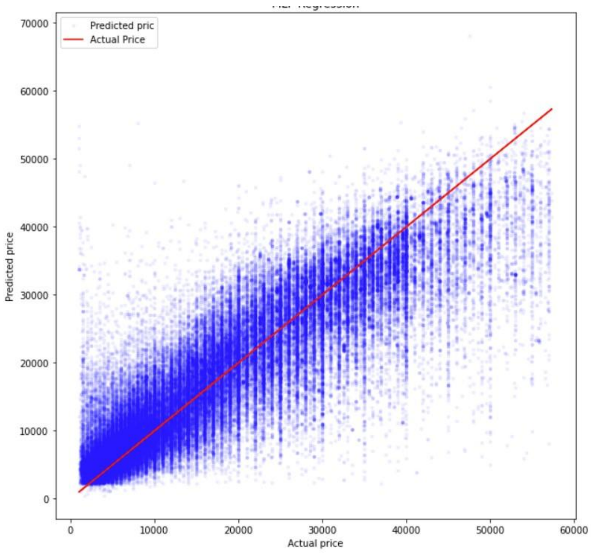


Fig. 21: Representation of the actual price and predicted price obtained through MLP Regression

8. Conclusion:

In this article, the dataset preprocessed which dataset which had nominal, ordinal, NA, unrelated and unique values. Iterative imputation is implemented to not remove valuable information from the dataset. Inter Quantile Range (IQR) method is used to remove outliers from the dataset.

We have compared with 10 different model algorithm approaches from Scikit-Learn to estimate the car price. Appropriate parameters are used to minimize the error and optimize the models.

In the table given below, we have shown the accuracy provided by each of the Regression Algorithms along with their R squared value, Mean Squared Error, and Absolute Mean Error.

The Table given below, i.e., Table 1 will give a summarized and a concise view of the results we have obtained through different Machine Learning Models so far.

Regression Algorithm	Root Mean Squared Error	R Squared	Absolute Mean Error	Median Error
Linear Regression	0.695954	0.517725	0.541383	0.443996
Ridge Regression	0.695954	0.517725	0.541383	0.443997
Lasso Regression	1.002183	-0.000064	0.837426	0.814684
Bayesian Regression	0.695954	0.517725	0.541384	0.444012
Decision Tree Regression	0.382053	0.854662	0.185816	0.049444
Random Forest Regression	0.270555	0.927114	0.14613	0.060435
Gradient Boost Regression	0.269204	0.92784	0.142366	0.058699
LGBM Regression	0.407457	0.834691	0.283383	0.193834
XGB Regression	**0.265757**	**0.929676**	**0.145751**	**0.064475**
MLP Regression	0.499533	0.751537	0.355409	0.248165

Table 1: Tabulation of Different Results produced by Different Regression Algorithms

Please note that the Best R squared value is produced by Random Forest Algorithm, and hence it has been highlighted.

The Bar graph representation of the R Squared Error for Different Regression Algorithms can be found below in Figure 22.

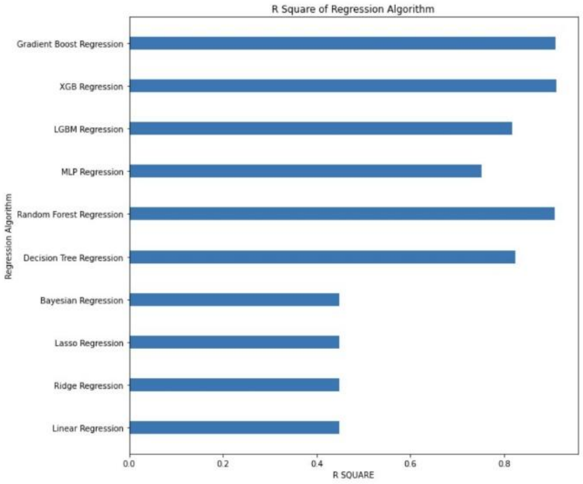


Fig. 22: Bar graph representation of R Squared Error

We have concluded that Linear regression, Ridge Regression, and Bayesian Regression give us an R squared score of 0.484352, while root mean squared value is 0.695954 since these models are basic and used linear equation to predict. Lasso Regression gives us an R squared score of -0.000064 and root mean squared value of 1.002183

When we come to the concept of Tree models, Random Forest has a R squared score of 0.927114 and the root mean squared value is 0.270555 followed by Decision Tree which has a R-squared score of 0.854462 accompanied by root mean squared error of 0.382053. Coming to the Artificial Neural Networks (ANN) models, Multi-Layer-Perception has R squared score of 0.751537 and the root mean squared value is 0.499533. We have the LGBM Regressor which has the least R-squared score of 0.834691 and the root mean squared value is 0.407457.

Extreme Gradient Boost Regressor has clearly the highest R squared score of 0.929676 and the lowest root mean squared error of 0.265757. Then, we have the Gradient Boosting Regressor with the R-squared score of 0.92784 and root mean squared error of 0.269204.

Hence, XGB Regressor is the most suitable model for predicting the price of the used cars since it shows more accuracy for the given input features. We have found that XGB Regression has the highest R-squared value. We have found that the R^2 value of XGB Regression is 0.929676, and hence we can say that it can predict the target variable with 92.96 % accuracy.

References:

[1] Ganesh, Mukkesh & Venkatasubbu, Pattabiraman. (2019). Used Cars Price Prediction using Supervised Learning Techniques. International Journal of Engineering and Advanced Technology. 9. 216-223. 10.35940/ijeat.A1042.1291S319.

[2] Mr. Ram Prashath R, NIthish C N, Ajith Kumar J.*"Price Prediction of Used Cars Using Machine Learning"*, Volume 10, Issue V, International Journal for Research in Applied Science and Engineering Technology (IJRASET) Page No: 4692-4695, ISSN : 2321-9653

[3] Praful Rane1, Deep Pandya2, Dhawal Kotak3 "Used car price prediction ";International Research Journal of Engineering and Technology (IRJET). (2021)

[4] Prof. Pallavi Bharambe, Bhargav Bagul, Shreyas Dandekar, Prerna Ingle.*"Used Car Price Prediction using Different Machine Learning Algorithms"*, Volume 10, Issue IV, International Journal for Research in Applied Science and Engineering Technology (IJRASET) Page No: 773-778, ISSN : 2321-9653, www.ijraset.com

[5] Gegic, Enis & Isakovic, Becir & Kečo, Dino & Mašetić, Zerina & Kevric, Jasmin. (2019). Car price prediction using machine learning techniques. TEM Journal. 8. 113-118. 10.18421/TEM81-16.

[6] F. Jaffar, T. Farid, M. Sajid, Y. Ayaz and M. J. Khan, "Prediction of Drag Force on Vehicles in a Platoon Configuration Using Machine Learning," in IEEE Access, vol. 8, pp. 201823-201834, 2020, doi: 10.1109/ACCESS.2020.3035318.

[7] Pudaruth, Sameerchand. (2014). Predicting the Price of Used Cars using Machine Learning Techniques. International Journal of Information & Computation Technology. 4. 753-764.

[8] Abhishek Jha, Dr. Ramveer Singh, Manish, Imran Saifi, Shipra Srivastava. Used car price prediction, International Journal of Advance Research, Ideas and Innovations in Technology, www.IJARIIT.com.

[9] C. Jin, "Price Prediction of Used Cars Using Machine Learning," 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT), 2021, pp. 223-230, doi: 10.1109/ICESIT53460.2021.9696839.