HOUSE PRICE PROJECT

Submitted by:
LAXMI NARAYAN

Table of Contents

Abstract

House price forecasting is an important topic of real estate. The literature attempts to derive useful knowledge from historical data of property markets. Machine learning techniques are applied to analyze historical property transactions in India to discover useful models for house buyers and sellers. Revealed is the high discrepancy between house prices in the most expensive and most affordable suburbs in the city of Mumbai. Moreover, experiments demonstrate that the Multiple Linear Regression that is based on mean squared error measurement is a competitive approach.

Introduction

Machine learning is a subfield of Artificial Intelligence (AI) that works with algorithms and technologies to extract useful information from data. Machine learning methods are appropriate in big data since attempting to manually process vast volumes of data would be impossible without the support of machines. Machine learning in computer science attempts to solve problems algorithmically rather than purely mathematically. Therefore, it is based on creating algorithms that permit the machine to learn. However, there are two general groups in machine learning which are supervised and unsupervised. Supervised is where the program gets trained on pre-determined set to be able to predict when a new data is given. Unsupervised is where the program tries to find the relationship and the hidden pattern between the data [1].

Several Machine Learning algorithms are used to solve problems in the real world today. However, some of them give better performance in certain circumstances, as stated in the No Free Lunch Theorem [2]. Thus, this thesis attempts to use regression algorithms and artificial neural network (ANN) to compare their performance when it comes to predicting values of a given dataset.

Aim And Purpose

The No Free Lunch Theorem state that algorithms perform differently when they are used under the same circumstances [2]. This study aims to analyse the accuracy of predicting house prices when using Multiple linear, Lasso, Ridge, Random Forest regression algorithms and Artificial neural network (ANN). Thus, the purpose of this study is to deepen the knowledge in regression methods in machine learning.

In addition, the given datasets should be processed to enhance performance, which is accomplished by identifying the necessary features by applying one of the selection methods to eliminate the unwanted variables since each house has its unique features that help to estimate its price. These features may or may not be shared with all houses, which means they do not have the same influence on the house pricing resulting in inaccurate output.

Background

2.1. Multiple Linear Regression

Multiple Linear Regression (MLR) is a supervised technique used to estimate the relationship between one dependent variable and more than one independent variables. Identifying the correlation and its cause-effect helps to make predictions by using these relations [4]. To estimate these relationships, the prediction accuracy of the model is essential; the complexity of the model is of more interest. However, Multiple Linear Regression is prone to many problems such as multicollinearity, noises, and overfitting, which effect on the prediction accuracy.

Regularised regression plays a significant part in Multiple Linear Regression because it helps to reduce variance at the cost of introducing some bias, avoid the overfitting problem and solve ordinary least squares (OLS) problems. There are two types of regularisation techniques L1 norm (least absolute deviations) and L2 norm (least squares). L1 and L2 have different cost functions regarding model complexity [5].

2.2. Lasso Regression

Least Absolute Shrinkage and Selection Operator (Lasso) is an L1-norm regularised regression technique that was formulated by Robert Tibshirani in 1996 [6]. Lasso is a powerful technique that performs regularisation and feature selection. Lasso introduces a bias term, but instead of squaring the slope like Ridge regression, the absolute value of the slope is added as a penalty term. Lasso is defined as:

$$L = Min(sum\ of\ squared\ residuals + \alpha * |slope|) \quad (1)$$

Where $Min(sum\ of\ squared\ residuals)$ is the Least Squared Error, and $\alpha * |slope|$ is the penalty term. However, alpha a is the tuning parameter which controls the strength of the penalty term. In other words, the tuning parameter is the value of shrinkage. $|slope|$ is the sum of the absolute value of the coefficients [7].

Cross-validation is a technique that is used to compare different machine learning algorithms in order to observe how these methods will perform in practice. Cross-validation method divides the data into blocks. Each block at a time will be used for testing by the algorithm, and the other blocks will be used for training the model. In the end, the results will be summarised, and the block that performs

best will be chosen as a testing block [8]. However, α is determined by using cross-validation. When $\alpha=0$, Lasso becomes Least Squared Error, and when $\alpha \neq 0$, the magnitudes are considered, and that leads to zero coefficients. However, there is a reverse relationship between alpha a and the upper bound of the sum of the coefficients t. When $t \to \infty$, the tuning parameter $a=0$. Vice versa when $t=0$ the coefficients shrink to zero and $a \to \infty$ [7]. Therefore, Lasso helps to assign zero weights to most redundant or irrelevant features in order to enhance the prediction accuracy and interpretability of the regression model.

Throughout the process of features selection, the variables that still have non-zero coefficients after the shrinking process are selected to be part of the regression model [7]. Therefore, Lasso is powerful when it comes to feature selection and reducing the overfitting.

2.3. Ridge Regression

The Ridge Regression is an L2-norm regularised regression technique that was introduced by Hoerl in 1962 [9]. It is an estimation procedure to manage collinearity without removing variables from the regression model. In multiple linear regression, the multicollinearity is a common problem that leads least square estimation to be unbiased, and its variances are far from the correct value. Therefore, by adding a degree of bias to the regression model, Ridge Regression reduces the standard errors, and it shrinks the least square coefficients towards the origin of the parameter space [10]. Ridge formula is:

$$R=Min(sum\ of\ squared\ residuals + \alpha * slope_2) \quad (2)$$

Where $Min(sum\ of\ squared\ residuals)$ is the Least Squared Error, and $\alpha * slope_2$ is the penalty term that Ridge adds to the Least Squared Error.

When Least Squared Error determines the values of parameters, it minimises the sum of squared residuals. However, when Ridge determines the values of parameters, it reduces the sum of squared residuals. It adds a penalty term, where α determines the severity of the penalty and the length of the slope. In addition, increasing the α makes the slope asymptotically close to zero. Like Lasso, α is determined by applying the Cross-validation method. Therefore, Ridge helps to reduce variance by shrinking parameters and make the prediction less sensitive.

2.4. Random Forest Regression

A Random Forest is an ensemble technique qualified for performing classification and regression tasks with the help of multiple decision trees and a method called Bootstrap Aggregation known as Bagging [11].

Decision Trees are used in classification and regression tasks, where the model (tree) is formed of nodes and branches. The tree starts with a root node, while the internal nodes correspond to an input attribute. The nodes that do not have children are called leaves, where each leaf performs the prediction of the output

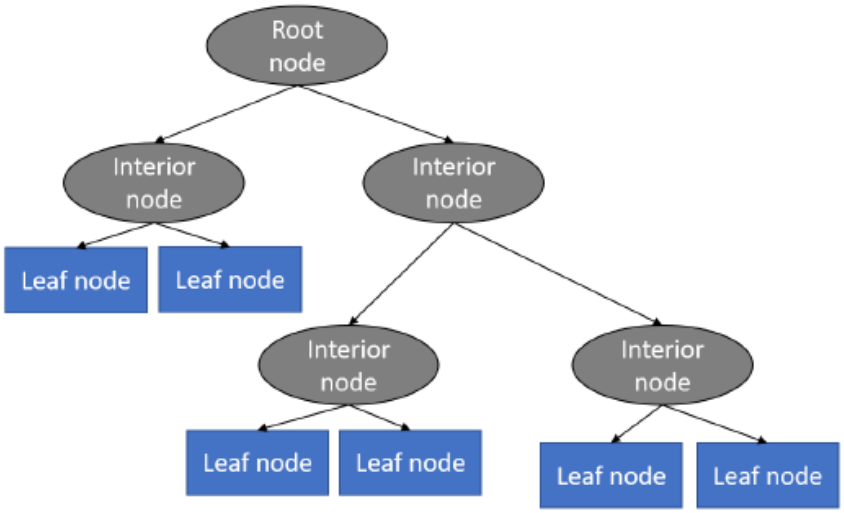


Figure 1. Decision Tree

A Decision Tree can be defined as a model [13]:

$$\varphi = X \mapsto Y \qquad (3)$$

Where any node t represents a subspace $X_t \subseteq X$ of the input space and internal nodes t are labelled with a split s_t taken from a set of questions Q. However, to determine the best separation in Decision Trees, the Impurity equation of dividing the nodes should be taken into consideration, which is defined as:

$$\Delta i(s,t) = i(t) - pLi(t_L) - pRi(t_R) \qquad (4)$$

Where $s \in Q$, t_L and t_R are left and right nodes, respectively. pL and pR are the proportion $N_{tL}N_t$ and $N_{tR}N_t$ respectively of learning samples from $\mathcal{L}_t$ going to t_L and t_R respectively. N_t is the size of the subset $\mathcal{L}_t$.

Random Forest is a model that constructs an ensemble predictor by averaging over a collection of decision trees. Therefore, it is called a forest, and there are two reasons for calling it random. The first reason is growing trees with a random independent bootstrap sample of the data. The second reason is splitting the nodes with arbitrary subsets of features [14]. However, using the bootstrapped sample and considering only a subset of the variables at each step results in a wide variety of trees. The variety is what makes Random Forest more effective than individual Decision Tree.

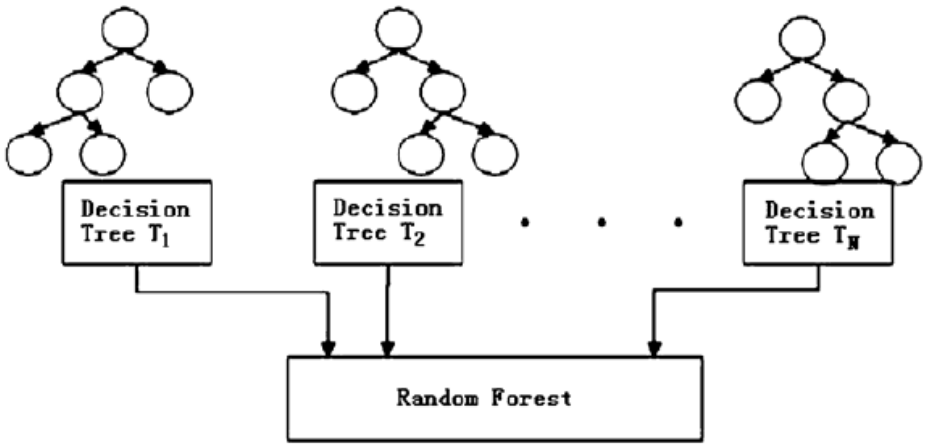


Figure 2. Random Forests

2.5. Artificial Neural Network

Artificial neural network (ANN) is an attempt to simulate the work of a biological brain. The brain learns and evolves through the experiments that it faces through time to make decisions and predict the result of particular actions. Thus, ANN tries to simulate the brain to learn the pattern in a given data to predict the output of that data whether the expected data was provided in the learning process or not [17].

ANN is based on an assemblage of connected elements or nodes called neurons. Neurons act as channels that take an input, process it, and then pass it to other neurons for further processing. This transaction or the process of transferring data between neurons is handled in layers. Layers consist of at least three layers, input layer, one or more of hidden layers and output layer. Each layer holds a set of neurons that takes input and process data and finally pass the output to other neurons in the next layer. This process is repetitive until the output layer has been reached,

So eventually, the result can be presented. ANN architecture is shown in the following figure as is also known as feed-forward, which values pass in one direction.

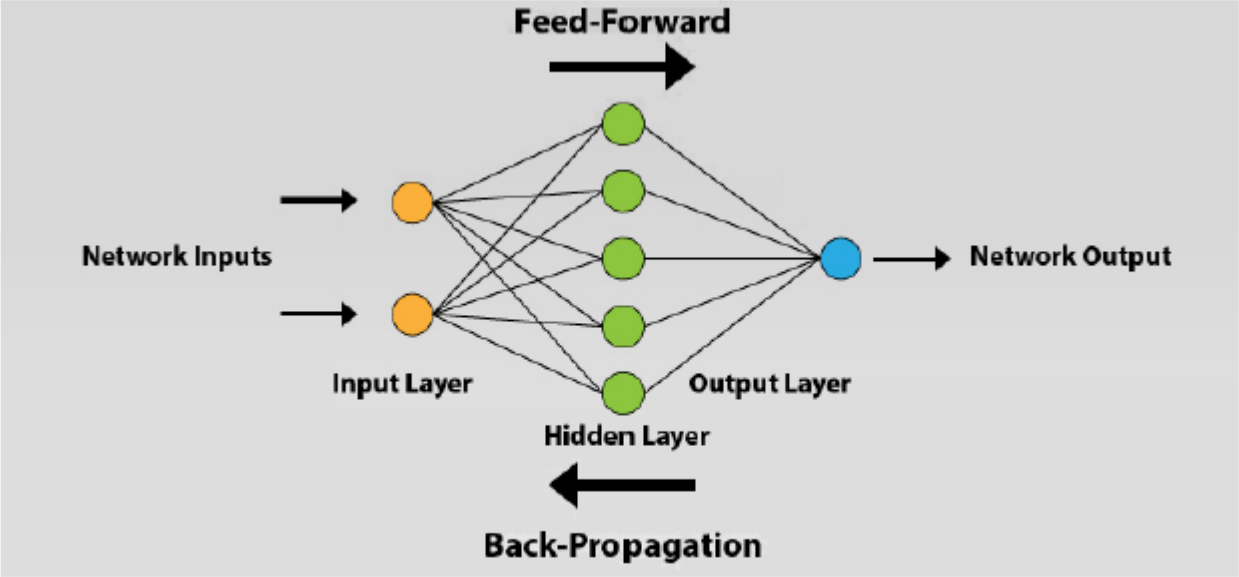


Figure 3. ANN architecture

.

The data that is being held in each neuron is called activation. Activation value ranges from 0 to 1. As shown in figure 3, each neuron is linked to all neurons in the previous layer. Together, all activations from the first layer will decide if the activation will be triggered or not, which is done by taking all activations from the first layer and compute their weighted sum [18].

$$w_1a_1+w_2a_2+w_3a_3+\cdots+w_na_n \quad (5)$$

However, the output could be any number when it should be only between 0 and 1. Thus, specifying the range of the output value to be within the accepted range. It can be done by using the Sigmoid function that will put the output to be ranging from 0 to 1. Then the bias is added for inactivity to the equation so it can limit the activation to when it is meaningfully active.

$$\sigma(w_1a_1+w_2a_2+w_3a_3+\cdots+w_na_n-b) \quad (6)$$

Where a is activation, w presents the weight, b is the bias and σ is the sigmoid function. Nevertheless, after getting the final activation, its predicted value needs to be compared with the actual value. The difference between these values is considered as an error, and it is calculated with the cost function. The cost function helps to detect the error percentage in the model, which needs to be reduced. Applying back-propagation on the model reduces the error percentage by running the procedures backwards to check on how the weight and bias are affecting the cost function.

2.6 Extreme Gradient Boosting (XGBoost)

XGBoost is a scalable machine learning system for tree boosting. The system is available as an open-source pack-age. The system has generated a significant impact and been widely recognized in various machine learning and data mining challenges [12]. The most crucial reason why XGBoost succeeds is its scalability in all scenarios. The system runs more than ten times faster than existing popular solutions on a single machine and scales to billions of examples in distributed or memory-limited settings. The scalability of XGBoost is due to several major systems and algorithmic optimizations including a novel tree learning algorithm for handling sparse data and a theoretically justified weighted quantile sketch procedure enabling instance weight handling in approximate tree learning. Parallel and distributed computing make learning faster, which allows quicker model exploration. More importantly, the model exploits out-of-

core computa-tion and enables data scientists to process a hundred millions of examples on a desktop. Finally, after combining these techniques to make an end-to-end system, it can scale to even more extensive data with the least amount of cluster resources [12]. In this paper, we utilized the XGBRegressor from xgboost open-source package [13]. After tweaking the XGBoost model multiple times, we set our parameter to the following:

• Set learning_rate = 0.1
• Set n_estimators = 200
• Determined the optimal tree specific parameters min_child_weight = 2, subsample = 1, colsample_bytre = 0.8
• Set requlariza tion parameter: reg_lambda = 0.45, reg_alpha = 0, gamma = 0.5
The model performed with a high accuracy where the RMSLE of the training set is around 0.16118.

3.1. Feature Engineering

Feature Engineering is the technique of improving the performance on a dataset by transforming its feature space, and it is the practice of constructing suitable features from given features of the dataset, which leads to improving the performance of the prediction model [25]. However, several techniques should be implemented for better performance and a prediction result [26].

3.1.1. Imputation

Missing value imputation is one of the biggest challenges encountered by the data scientist. In addition, most machine learning algorithms are not powerful enough to handle missing data. Missing data can lead to ambiguity, misleading conclusions, and results [27]. There are two types of missing values [28]; the first type is called missing completely at random (MCAR). MCAR can be expressed as:

$$P(R|X,Z,\mu)=P(R|\mu) \quad (7)$$

Where R is the response indicator variables, X are independent of data variables, and Z is latent. The second type is called missing at random (MAR), which can be expressed as:

$$P(R=r|X=x,Z=z,\mu)=P(R=r|X_0=x_0) \ (8) \ for\ all\ x_\mu,z\ and\ \mu \quad (9)$$

There are two methods of handling missing data, namely ignoring missing data and imputation of missing data. Ignoring missing data is a simple technique which deletes the cases that contain missing data. The disadvantages of this method are that it reduces the size of the dataset, and it uses a different sample size for different variables. Imputation of missing data is a technique that replaces missing data with some reasonable data values [27]. However, the imputation of missing data method has two types, single imputation, and multiple imputations.

Single imputation contains several approaches, such as mean imputation and regression imputation. Mean imputation is the most common approach of missing data replacement [27]. It replaces the missing data with sample mean or median. However, it has a disadvantage which is if missing data are enormous in number, then all those data are replaced with the same imputation mean, which leads to change in the shape of the distribution. Regression imputation is a technique based on the assumption of the linear relationship between the attributes. The advantage of regression imputation over mean imputation is that it was able to preserve the distribution shape [27].

3.1.2. Outliers

Outliers are noisy data that they do have abnormal behaviour comparing with the rest of the data in the same dataset. Outliers can influence the prediction model and performance due to its oddity. There are three types of outliers, which are point, contextual, and collective outliers [29]. Point outlier is an individual data instance that can be considered as odd with respect to the rest of the data. The contextual outlier is an instance of data that can be regarded as odd in a specific context but not otherwise. An example of contextual is the longitude of a location. A collective outlier is a collection

of related data instances that can be considered as abnormal with respect to the entire dataset. In supervised, the detection of outliers can be accomplished visually, where a predictive model is built for normal against outliers' classes. Dean De has investigated the public dataset and he suggests to remove certain outliers from the public data when he said "I would recommend removing any houses with more than 4000 square feet from the data set" [30]. Another example of detecting outliers is by using Isolation forest, which has two stages, training, and testing. The training is to create the isolation trees and then to record the anomaly score of each entry in the testing stage. This method has shown a promising result, according to [31].

3.1.3. Log Transformation

A log transformation is a method that is used to handle skewed data. It is used to make data conform to normality, it reduces the impact of the outliers, due to the normalisation of magnitude and to reduce the variability of data [33].

3.1.4. One-hot Encoding

One-hot encoding is a technique that is used to convert categorical features to a suitable format to be used as an input in Machine Learning algorithms [34]. It transforms a single variable with n observations and d distinct values to d binary variables, where each observation indicating the presence as 1 or absence as 0 [35]. In one-hot encoding, the categories are represented as independent concepts.

3.1.5. Feature Selection

Feature Selection is an important technique that is used to handle high-dimensional input data and overfitting caused by a curse of dimensionality by selecting a relevant feature subset based on mutual information criterion [36]. Moreover, feature selection has many advantages, such as improve the prediction performance by reducing dimensionality in the dataset. It speeds up the learning process

and leads to a better understanding of the considered problem. However, there are many useful methods for feature selection, such as Mutual Information (MI) and Conditional Mutual Information (CMI) [37]. Mutual information is used for quantifying the mutual dependence of random variables, and it can be considered as the amount of information shared by two variables. MI is given as:

$$I(X;Y)=\Sigma\Sigma p(xy)logp(xy)p(x)p(y)\ _{y\in Y x\in X} \quad (12)$$

Where $x\in X$ and $y\in Y$ are the possible value assignments of X and Y, and log is used base 2. Conditional Mutual Information measures the limited dependence between two random variables given the third [37].

$$I(X;Y|Z)=\Sigma p(z)_{z\in Z}\Sigma\Sigma p(xy|z)logp(xy|z)p(x|z)p(y|z)\ _{y\in Y x\in X} \quad (13)$$

4.1. Evaluation Metrics

Several evaluation metrics measure the performance of machine learning algorithms such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R-Squared, and Mean Absolute Error (MAE). However, in this study, the performance of the algorithms is measured by using RMSE and R-Squared.

Root Mean Square Error (RMSE) is used as an evaluation metric in machine learning to measure the performance of the model. However, RMSE is similar to the Mean Square Error (MAE). Where all errors in MAE have the same weight, but RMSE penalises the variance, which means it gives more weight to the errors that have large absolute values than that have small absolute values.

Therefore, when RMSE and MAE are calculated, RMSE is always bigger than MAE. RMSE is more sensitive to the errors than MAE; therefore, using RMSE for measuring the performance is better than MAE [38]. RMSE can be calculated as the square root of the sum of squared errors$\Sigma(y_i-\hat{y_i})_{2ni}$ over the sample size n. RMSE can be presented as:

4.2. Correlation

Correlation analysis defines the strength of a relationship between two variables, which can be between two independent variables or one independent and one dependent variable. The strength of the relationship can be distinguished based on direction and dispersion strength as in figure 4.

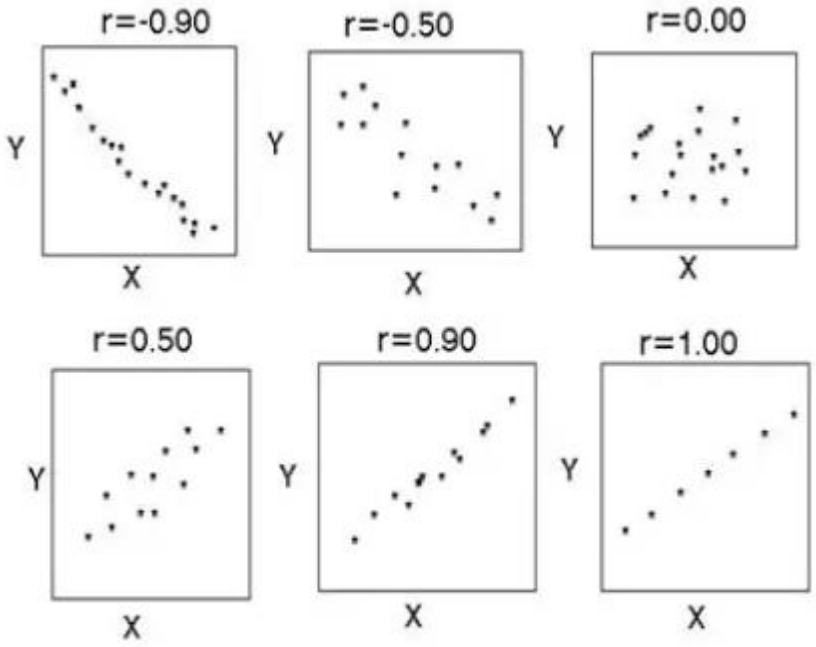


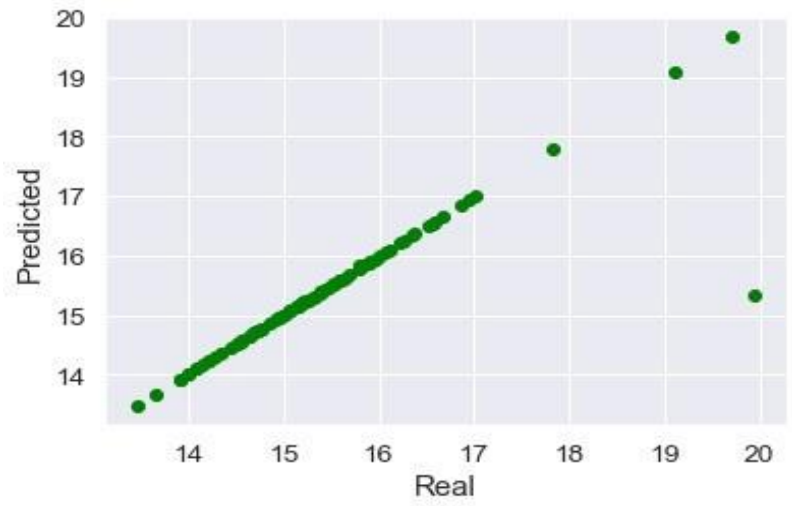
Figure 4. Correlation strength of the value of R

MODELS USED

Regression Model

- Linear Regression is a machine learning algorithm based on supervised learning.
- It performs a regression task. Regression models a target prediction value based on independent variables.
- It is mostly used for finding out the relationship between variables and forecasting.

Real Vs Predicted

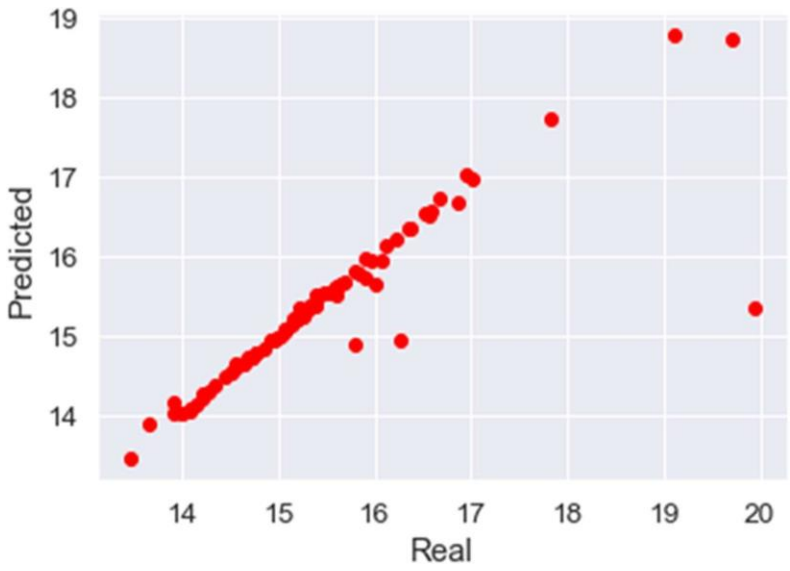


Random Forest Regression Model

- A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap Aggregation, commonly known as bagging.

- Bagging, in the Random Forest method, involves training each decision tree on a different data sample where sampling is done with replacement.

- The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.

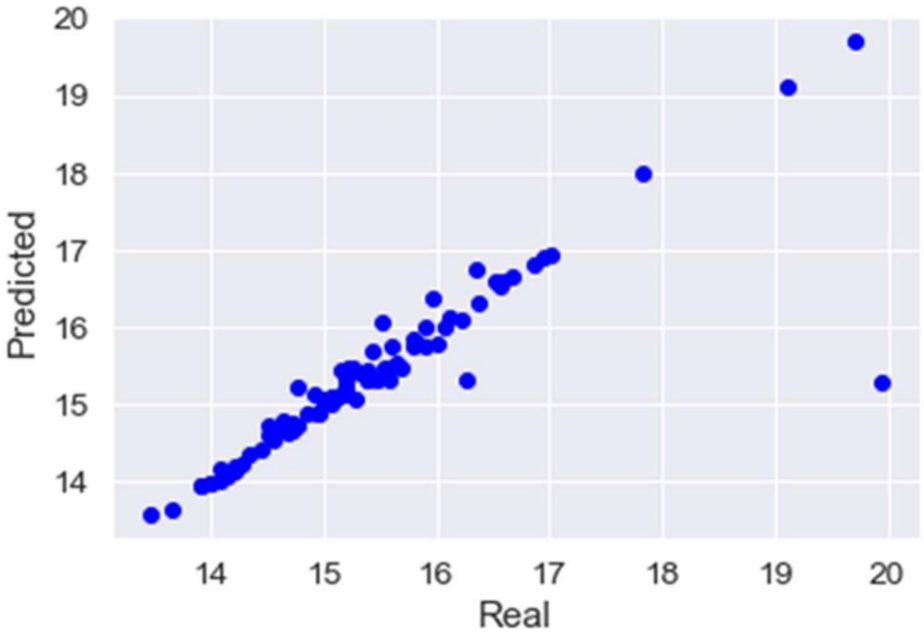


Real Vs Predicted

XG Boost Regressor Model

- XG Boost stands for eXtreme Gradient Boosting.
- The XG Boost library implements the gradient boosting decision tree algorithm.
- Boosting is an ensemble technique where new models are added to correct the errors made by existing models.
- Models are added sequentially until no further improvements can be made.

Real Vs Predicted



5.Implementation

Dataset: T he public dataset is taken from a website called Kaggle.

```
In [92]: house_price.head()
```

Out[92]:

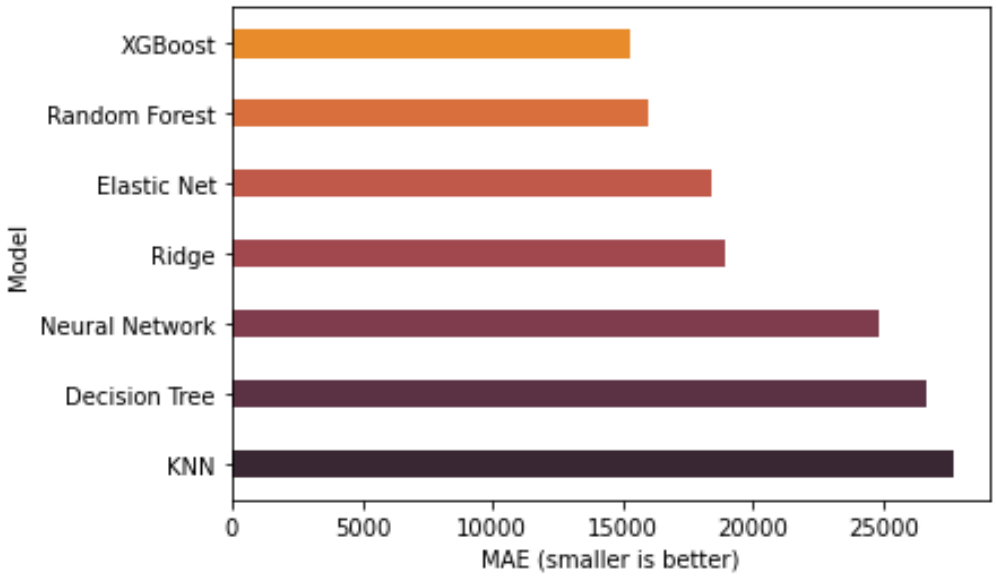
	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSc
0	127	120	RL	NaN	4928	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
1	889	20	RL	95.0	15865	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
2	793	60	RL	92.0	9920	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
3	110	20	RL	105.0	11751	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	
4	422	20	RL	NaN	16635	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	

5 rows × 81 columns

Experiment Results

Many machine learning algorithms are used to predict. However, previous researches have shown a comparison between them alongside Artificial neural network in different datasets. Therefore, using these algorithms is beneficial so that the result can be as near to the claimed results. However, the prediction accuracy of these algorithms depends heavily on the given data when training the model. If the data is in bad shape, the model will be overfitted and inefficient, which means that data pre-processing is an important part of this experiment and will affect the final results. Thus, multiple combinations of pre-processing methods need to be tested before getting the data ready to be used in training.

The practical results show that Lasso is the most accurate algorithm among other algorithms,



6.Conclusion

We have managed out how to prepare a model that gives users for a novel best approach with take a gander at future lodging value predictions. A few relapse strategies have been investigated Furthermore compared, when arriving during a prediction strategy In light of XG support. Straight former imply works bring been utilized within our model, something like that that future value predictions will have a tendency towards All the more sensible value.

References

- Alkhatib, K., Najadat, H., Hmeidi, I., & Shatnawi, M. K. A. (2013). Stock price prediction using k-nearest neighbor (kNN) algorithm. International Journal of Business, Humanities and Technology, 3(3), 32-44.

- de Abril, I. M., & Sugiyama, M. (2013). Winning the kaggle algorithmic trading challenge with the composition of many models and feature engineering. IEICE transactions on information and systems, 96(3), 742-745.

- Feng, Y., & Jones, K. (2015, July). Comparing multilevel modelling and artificial neural networks in house price prediction. In Spatial Data Mining and Geographical Knowledge Services (ICSDM), 2015 2nd IEEE International Conference on (pp. 108-114). IEEE.

- Hegazy, O., Soliman, O. S., & Salam, M. A. (2014). A machine learning model for stock market prediction. arXiv preprint arXiv:1402.7351.

- Ticknor, J. L. (2013). A Bayesian regularized artificial neural network for stock market forecasting. Expert Systems with Applications, 40(14), 5501-5506.

- De Cock, D. (2011). Ames, Iowa: Alternative to the Boston housing data as an end of semester regression project. Journal of Statistics Education, 19(3).

- www.kaggel.com

- Annina S, Mahima SD, Ramesh B. An Overview of Machine Learning and its Applications. International Journal of Electrical Sciences & Engineering (IJESE). 2015 January; I(1): 22-24.

- Fonti V. Feature Selection using LASSO. VU Amsterdam Research Paper in Business Analytics. 2017 Mars: p. 1-25.

- David HW, William GM. No Free Lunch Theorems for Optimisation. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. 1997 April; I(1): 67-82.