International Journal for Modern Trends in Science and Technology, 6(12): 103-107, 2020
Copyright © 2020 International Journal for Modern Trends in Science and Technology
ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST061220
Available online at: http://www.ijmtst.com/vol6issue12.html

A Comparative Study on House Price Prediction

Akash Dagar[1] | Shreya Kapoor[2]

[1]Student, Maharaja Agrasen Institute of Technology, Delhi, India
[2]Assistant Professor, Maharaja Agrasen Institute of Technology, Delhi, India

To Cite this Article
Akash Dagar and Shreya Kapoor, "A Comparative Study on House Price Prediction", *International Journal for Modern Trends in Science and Technology*, 6(12): 103-107, 2020.

Article Info
Received on 06-November-2020, Revised on 25-November-2020, Accepted on 29-November-2020, Published on 02-December-2020.

ABSTRACT

Machine learning plays a major role from past years in image detection, spam reorganization, normal speech command, product recommendation and medical diagnosis. Present machine learning algorithm helps us in enhancing security alerts, ensuring public safety and improve medical enhancements. Due to increase in urbanization, there is an increase in demand for renting houses and purchasing houses. Therefore, to determine a more effective way to calculate house price accurately is the need of the hour. So, an effort has been made to determine the most accurate way of predicting house price by using machine learning algorithms: Multivariable Linear Regression, Decision Tree Regression and Random Forest Regression and it is determined that Multivariable Linear Regression has showed most accuracy and less error.

KEYWORDS: *Machine Learning, Multivariable Linear Regression, Decision Tree Regression and Random Forest Regression, House Price Prediction*

I. Introduction

We all know that real estate property has always been a basic need for an individual. Due to this, every single organization in real estate business is trying to get an edge over other, which result in false price of the house, most of the time. As the population is also increasing day by day, so there arises a need to have an effective way to calculate or to predict the price of house. The buyers and the sellers, both are affected by the inconsistent and false prices, so the solution to this problem will give an idea about the price of a house on the basis of its features to both of them.

Regression is best for prediction like these. Regression is a machine learning apparatus that encourages you to make expectations by taking in – from the current measurable information – the connections between your target parameter and a lot of different independent parameters. For the proposed model various regression algorithms like Multivariable Linear regression algorithm, Decision Tree regression and Random Forest regression are compared to get the most accurate results.

DATASET

The dataset used here is the real dataset. It's an Indian dataset of state Bangalore which has 9 columns and around 13000 rows. There are 8 independent variables and one target variable i.e. price.

Variables of Dataset are as follows:
1. Area Type – describes the area
2. Availability – when it can be possessed or when it is ready (categorical and time-series)
3. Location – where it is located in Bengaluru (Area name)
4. Size – in BHK or Bedroom (1-10 or more)
5. Society – to which society it belongs

6. Total Square Feet – size of the property in square feet
7. Bath – No. of bathrooms
8. Balcony – No. of the balcony
9. Price – price of the house

II. RELATED WORK

Related Work or Literature survey is the most important step in any kind of research. Before start developing we need to study the previous papers of our domain which we are working and on the basis of study we can predict or generate the drawback and start working with the reference of previous papers. In this section, we briefly review the related work on house price prediction and the techniques used. There are many factors that have an impact on house prices, such as the number of bedrooms and bathrooms. House price depends upon its location as well. A house with great accessibility to highways, schools, malls, employment opportunities, would have a greater price as compared to a house with no such accessibility. Predicting house prices manually is a difficult task and generally not very accurate, hence there are many systems developed for house price prediction. Sifei Lu, Zengxiang Li, Zheng Qin, Xulei Yang, Rick Siow Mong Goh [1] had proposed an advanced house prediction system using linear regression. This system's aim was to make a model that can give us a good house price prediction based on other variables. They used the Linear Regression for Ames dataset and hence it gave good accuracy. The house price prediction project had two modules namely, Admin and the User. Admin can add location and view the location. Admin had the authority to add density on the basis of per unit area. Users can view the location and see the predicted housing price for that particular location. This paper [1] proposed on Hybrid Regression technique for housing Prices Prediction focused on the use of creative feature engineering to find the optimal features and their correlation with Sales Prices. Feature engineering improved the data normality and linearity of data. Their system showed that working on the Ames Housing dataset was convenient and showed that the use of Hybrid algorithms (65% Lasso and 35% Gradient Boost) provided results in predicting the house prices rather than using one from lasso, ridge or gradient boost.

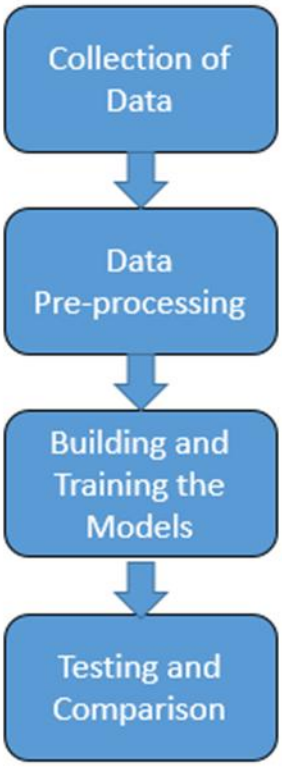
The paper proposed by Ayush Varma Abhijit Sharma Sagar Doshi Rohini Nair [2] suggested that the use of neural networks along with linear and boosted algorithms improved prediction accuracy. The dataset used here contained various essential parameters.The dataset was cleaned up. Three algorithms were used namely Linear Regression, Forest Regression and Boosted Regression. The dataset was tested on all three and the results of all the above algorithms were fed as an input to the neural network. Neural networks were used mainly to compare all the predictions and display the most accurate result. A neural network along with Boosted Regression was used to increase the accuracy of the result.

The paper proposed by Adyan Nur Alfiyatin, Hilman Taufiq, Ruth Ema Febrita, Wayan Firdaus Mahmudy [3] shows the prediction model is based on regression analysis and particle swarm optimization(PSO). Hedonic pricing is implemented using regression techniques to predict the NJOP price (Dependent Variable) in the city of Malang, based on factors such as land area, NJOP land price, NJOP building price. PSO is a stochastic optimization technique used for the selection of affect variables. The results obtained show a minimum prediction error RMSE of 14.186.

The paper proposed by Neelam Shinde, Kiran Gawande [4] surveyed to predict a continuous target value, using algorithms Logistic Regression, Support Vector Machine, Lasso Regression Technique and Decision Tree are used to build a predictive model. They have used a stepwise approach from Data Collection, Data Processing, Data Analysis, to Evaluating Models. Then the predicted output is stored in a CSV file. It was found that the Decision Tree had the best accuracy of 84% approx., they tried to implement the problem of Regression using the Classification Algorithm which was successful. They had used predefined open source Kaggle Dataset consisting of 80 parameters, from which 37 parameters were chosen which were affecting house prices.

III. METHODOLOGY

The methodology will go according to the following flowchart.



- Collection of Data

Data is the heart of machine learning. Predictive models use data for training which gives somewhat accurate results. Without data we can't train the model. Machine learning involves building these models from data and uses them to predict new data. Machine Learning is a subset of Artificial Intelligence. It gives system capability to learn wherein it automatically learns and improves its performance without being explicitly programmed. The dataset used here is taken from Kaggle.

- Data Pre-processing

Data pre-processing is the process of cleaning our data set. There might be missing values in the dataset or the values of the features can be oddly distributed. These can be handled by data cleaning. If there are missing values in a variable we will drop those values or substitute it with the average value and if the number of missing values is big then we have to drop that variable from our dataset. Some of the columns such as society, area type, balconies and availability were dropped of the dataset as society column has a lot of missing values and the others have not an important part in predicting the house price. A new feature was engineered and outliers were removed by using different methods. One Hot Encoding was used for categorical data.

- Building and Training the model

The dataset was split into training and testing set with 80% in training set and remaining 20% data in test set.

3 Models were used:

- Multivariable Linear Regression Model (MLR)

Also known as Multiple Linear Regression is the advanced form of Linear Regression, it uses more than one independent variables to find the dependent variable unlike linear regression where, one independent variable is used to predict one dependent variable. In MLR, the independent variables are not highly correlated to each other. MLR is used extensively in econometrics and financial inference.

- Decision Tree Regression Model (DTR)

Decision Tree Regression, as the name suggests it uses tree like structure to build regression and classification models. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches, each representing values for the attribute tested. Leaf node represents a decision on the numerical target. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

- Random Forest Regression Model (RFR)

Random forest is a Supervised Learning algorithm which uses ensemble learning method for classification and regression.
Random forest is a bagging technique and not a boosting technique. The trees in random forests are run in parallel. There is no interaction between these trees while building the trees.
It operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

Scikit-learn library was used for importing the models. Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. The vision for the library is a level of robustness and support required for use in production systems. This means a deep focus on concerns such as easy of use, code quality, collaboration, documentation and performance.

K-fold Cross Validation is used for training and testing the models accurately. Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample.

The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation. When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as k=10 becoming 10-fold cross-validation.

Cross-validation is primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data. That is, to use a limited sample in order to estimate how the model is expected to perform in general when used to make predictions on data not used during the training of the model. It is a popular method because it is simple to understand and because it generally results in a less biased or less optimistic estimate of the model skill than other methods, such as a simple train/test split.

- Testing and Comparison

Testing was done using K-Fold Cross-Validation. Root Mean Square Error is used for comparing the models.

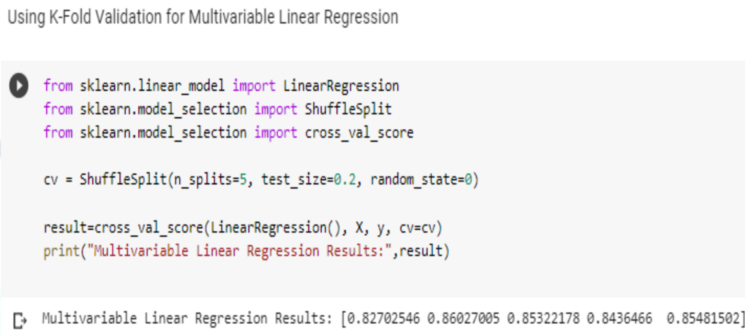
Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. Root mean square error is commonly used in climatology, forecasting, and regression analysis to verify experimental results.

Formula is:

Where:
f = forecasts (expected values or unknown results),
o = observed values (known results).

IV. Results

With the help of K-Fold Cross Validation we have the following accuracy results for different models.



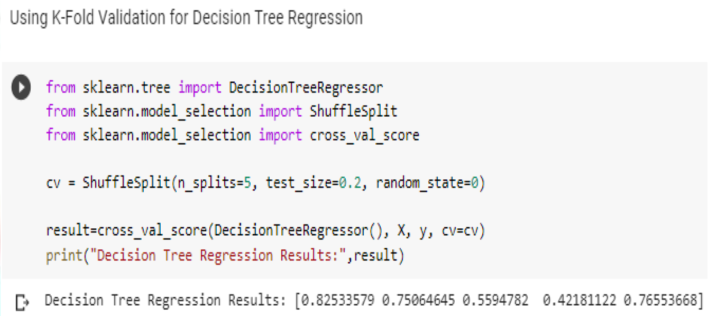
Using K-Fold Validation for Multivariable Linear Regression

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score

cv = ShuffleSplit(n_splits=5, test_size=0.2, random_state=0)

result=cross_val_score(LinearRegression(), X, y, cv=cv)
print("Multivariable Linear Regression Results:",result)
```

```
Multivariable Linear Regression Results: [0.82702546 0.86027005 0.85322178 0.8436466  0.85481502]
```



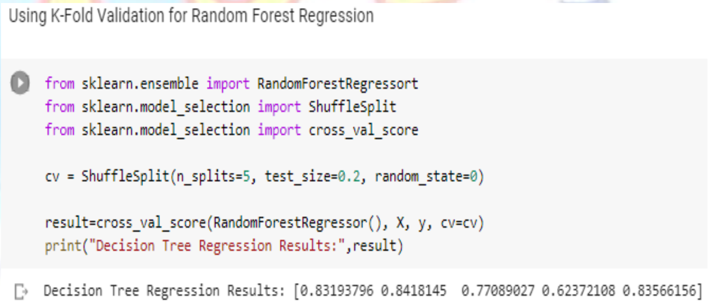
Using K-Fold Validation for Decision Tree Regression

```
from sklearn.tree import DecisionTreeRegressor
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score

cv = ShuffleSplit(n_splits=5, test_size=0.2, random_state=0)

result=cross_val_score(DecisionTreeRegressor(), X, y, cv=cv)
print("Decision Tree Regression Results:",result)
```

```
Decision Tree Regression Results: [0.82533579 0.75064645 0.5594782  0.42181122 0.76553668]
```



Using K-Fold Validation for Random Forest Regression

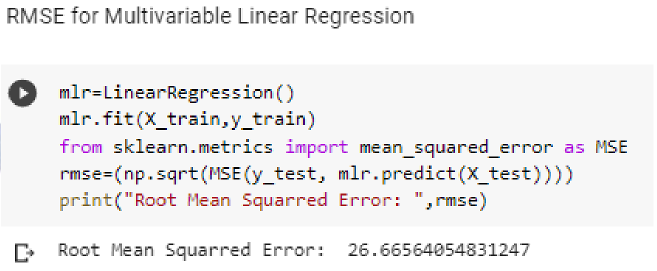
```
from sklearn.ensemble import RandomForestRegressort
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score

cv = ShuffleSplit(n_splits=5, test_size=0.2, random_state=0)

result=cross_val_score(RandomForestRegressor(), X, y, cv=cv)
print("Decision Tree Regression Results:",result)
```

```
Decision Tree Regression Results: [0.83193796 0.8418145  0.77089027 0.62372108 0.83566156]
```

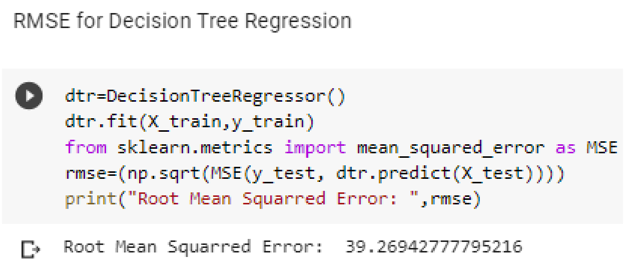
With the help of Root Mean Square Error we have the following errors for different models.



RMSE for Multivariable Linear Regression

```
mlr=LinearRegression()
mlr.fit(X_train,y_train)
from sklearn.metrics import mean_squared_error as MSE
rmse=(np.sqrt(MSE(y_test, mlr.predict(X_test))))
print("Root Mean Squarred Error: ",rmse)
```

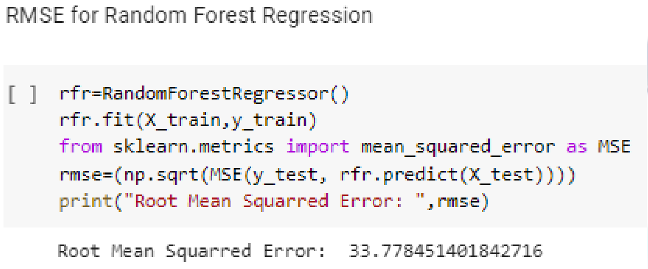
```
Root Mean Squarred Error:  26.66564054831247
```



RMSE for Decision Tree Regression

```
dtr=DecisionTreeRegressor()
dtr.fit(X_train,y_train)
from sklearn.metrics import mean_squared_error as MSE
rmse=(np.sqrt(MSE(y_test, dtr.predict(X_test))))
print("Root Mean Squarred Error: ",rmse)
```

```
Root Mean Squarred Error:  39.26942777795216
```



RMSE for Random Forest Regression

```
[ ] rfr=RandomForestRegressor()
    rfr.fit(X_train,y_train)
    from sklearn.metrics import mean_squared_error as MSE
    rmse=(np.sqrt(MSE(y_test, rfr.predict(X_test))))
    print("Root Mean Squarred Error: ",rmse)
```

```
Root Mean Squarred Error:  33.778451401842716
```

The following table represents the accuracy and error of different models.

Models	RMSE	Accuracy using K-fold Cross Validation
MLR	26.66564054831247	[0.82702546 0.86027005 0.85322178 0.8436466 0.85481502]
DTR	39.26942777795216	[0.82533579 0.75064645 0.5594782 0.42181122 0.76553668]
RFR	33.778451401842716	[0.83193796 0.8418145 0.77089027 0.62372108 0.83566156]

V. Conclusion

In this research paper, we have used machine learning algorithms to predict the house prices. The data is pre-processed and fed to different regression models to determine the best model and with the help of both K-Fold Cross-Validation and RMSE we found the accuracy of different models. So, according to above results we came to know that Multivariable Linear Regression is the most effective model with maximum accuracy and lowest RMSE. And can be used as an effective model for predicting house prices, which should profit both buyers and sellers.

Acknowledgment

I thank my Mentor Ms. Shreya Kapoor for advising on all aspects related to this paper, along with other concepts too. Later for validating the experimental result and reviewed the paper.

References

[1] Sifei Lu, Zengxiang Li, Zheng Qin, Xulei Yang, Rick Siow Mong Goh

[2] Ayush Varma Abhijit Sharma Sagar Doshi Rohini Nair

[3] Adyan Nur Alfiyatin, Hilman Taufiq, Ruth Ema Febrita, Wayan Firdaus Mahmudy

[4] Valuation Of House Prices Using Predictive Techniques NEELAM SHINDE, KIRAN GAWANDE Volume-5, Issue-6, Jun.-2018

[5] House Price Forecasting Using Machine Learning Alisha Kuvalekar Shivani Manchewar Sidhika Mahadik Shila Jawale (GUIDE) SSRN

[6] Housing Price Prediction using Machine Learning Yashraj Garud , Hemanshu Vispute , Nayan Bisai and Prof. Madhu Nashipudimath4 Volume: 07 Issue: 05 | May 2020 (IRJET)

[7] HOUSE COST PREDICTION USING DATA SCIENCE AND MACHINE LEARNING Anuj V. Kumar1, Anshuman Kumar2, Satish S. Tiwari 3, Sneha G. Gobade4, Prof. Amita Suke5 Volume: 07 Issue: 01 | Jan 2020.