House Price Prediction Using Ensembled

Machine Learning Model

Jeevan Mallik S  
Computer Science  
REVA University   
Bengaluru, India  
[jeevanmallik47@gmail.com](mailto:jeevanmallik47@gmail.com)Dhananjaya D V  
Computer Science  
REVA University   
Bengaluru, India  
[dhananjayadhana652@gmail.com](mailto:dhananjayadhana652@gmail.com)

Gangadhar Hakkapakki  
Computer Science  
REVA University  
Bengaluru, India  
[saigangadharhakkapakki23@gmail.com](mailto:saigangadharhakkapakki23@gmail.com)Mallikarjun  
Computer Science  
REVA University  
Bengaluru, India  
[mallikarjunmallikarjun096@gmail.com](mailto:mallikarjunmallikarjun096@gmail.com)

*Abstract* — This paper presents a comparison between two well-known machine learning algorithms, linear regression and decision tree regression, in their ability to predict house prices. The dataset used in this study includes various attributes of residential properties, such as the number of bedrooms, bathrooms, and square footage, among others. The performance of the algorithms is evaluated using commonly used metrics such as mean squared error and mean absolute error. The findings show that linear regression and decision-tree regression are effective in predicting house prices, with decision- tree regression performing slightly better in accuracy. Overall, this study Demo proves potential of machine learning algorithms for real estate valuation and offers insights into their comparative performance in this domain.

Keywords — House Price Prediction, Decision Tree, Linear Regression, Machine Learning, Feature Importance, Real Estate Valuation.

# Introduction

Predicting house prices accurately is a challenge that the real estate industry faces due to various economic factors that can cause fluctuations in the housing market. To address this, we conducted a study using two machine learning algorithms - Linear Regression and Decision Tree Regression [1]. We collected data from various sources and applied feature selection techniques to decide the most important variables in predicting house prices. Our findings show that both algorithms are effective in predicting house prices, with Decision Tree Regression performing slightly better in terms of accuracy. We used commonly used metrics like mean squared error and mean absolute error to evaluate the performance of the models [2]. Additionally, we found that the number of bedrooms and square footage are the most crucial factors in predicting house prices. Our study proves the potential of machine learning algorithms in real estate valuation and offers insights into their comparative performance in this domain.

# Related Work

Shelter is a basic human need, and owning a house is a significant financial investment. Unfortunately, many people make mistakes when dealing with properties. Agreeing to a deal without understanding the actual value of a property can led to financial consequences. Our goal is to create a prototype that can benefit the real estate industry. One powerful tool we use to analyze data is machine learning (ML). ML models have been developed across various domains due to their flexibility, including for predicting housing prices. Past research has implemented ML algorithms on housing datasets to make predictions of several types [sfgsj]. This study aims to develop a reliable prediction model for housing prices using machine learning algorithms.

## Machine Learning

Machine learning (ML) is an area of study that focuses on the development of algorithms and statistical models used by computer systems to perform tasks without explicit programming. The main goal of machine learning is to ease machines in learning from data. The process of machine learning involves training a machine learning model on a large dataset to recognize patterns and relationships in the data. The machine learning model then applies these patterns to new, unseen data to make accurate predictions or classifications. The goal of machine learning is to create intelligent machines that can learn from data, adapt to new situations, and make decisions or predictions with minimal human intervention. Researchers have conducted many studies on how to make machines learn without explicit programming. This has led mathematicians and programmers to explore various approaches for finding solutions to this challenge, particularly when dealing with large data sets [djfewhuf].

## *Supervised Learning in Machine Learning*

Supervised machine learning involves feeding labeled data into a machine learning model. The model is trained using input and output data with known values, allowing it to predict future outputs accordingly [dsfhdv].

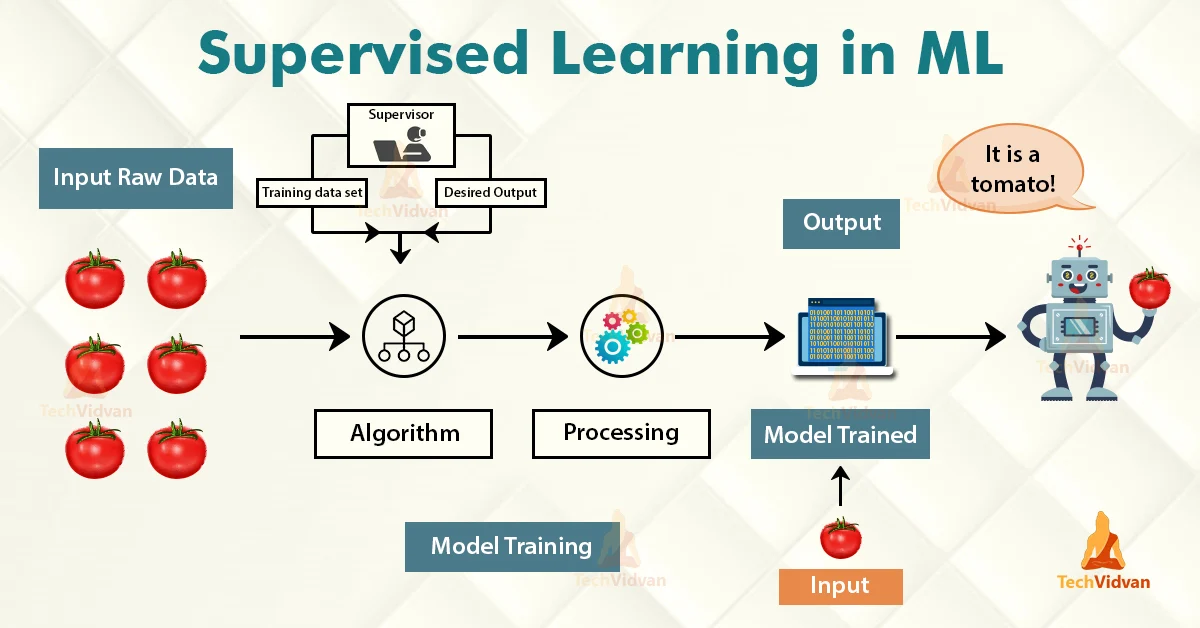


Fig 1. Supervised Machine Learning

## Unsupervised Learning in ML

Unsupervised learning is a machine learning method that does not require explicit direction for the model. This approach mainly deals with unlabeled data. Some examples of unsupervised learning algorithms include anomaly detection, clustering, and neural networks [dvkdkv].

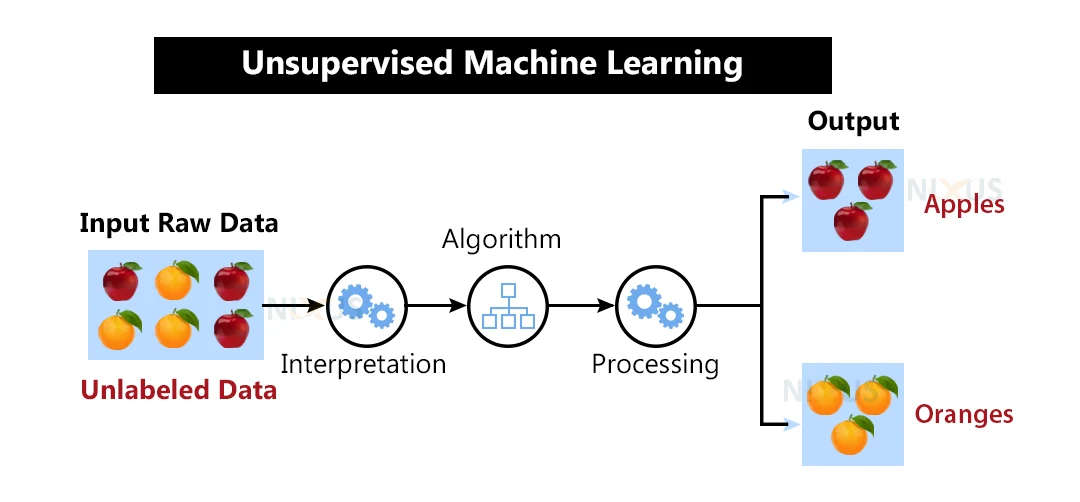


Fig 2. Unsupervised Machine Learning

## Prediction in ML

Prediction is the result of using an algorithm that has been trained on past data and then applied to new data to estimate the probability of a specific outcome. By using machine learning models, we can make highly accurate predictions based on historical data [sdvkjdkv]. There are several machine learning algorithms available for prediction, including linear regression, multiple linear regression, random forest, regression tree, and neural networks.

# Algorithm That We Have Used

## Linear Regression:

* Linear regression is a commonly used approach in supervised machine learning that involves predicting the value of a dependent variable (Y) based on a given independent variable (X). The aim of linear regression is to establish a linear relationship between the two variables and use it to make predictions about Y for new values of X. The relationship between the variables can be expressed using the Equation:
* Y = mX + b
  + Y is dependent variable.
  + X is an independent variable.
  + m is an estimated slope.
  + b is estimated intercept.

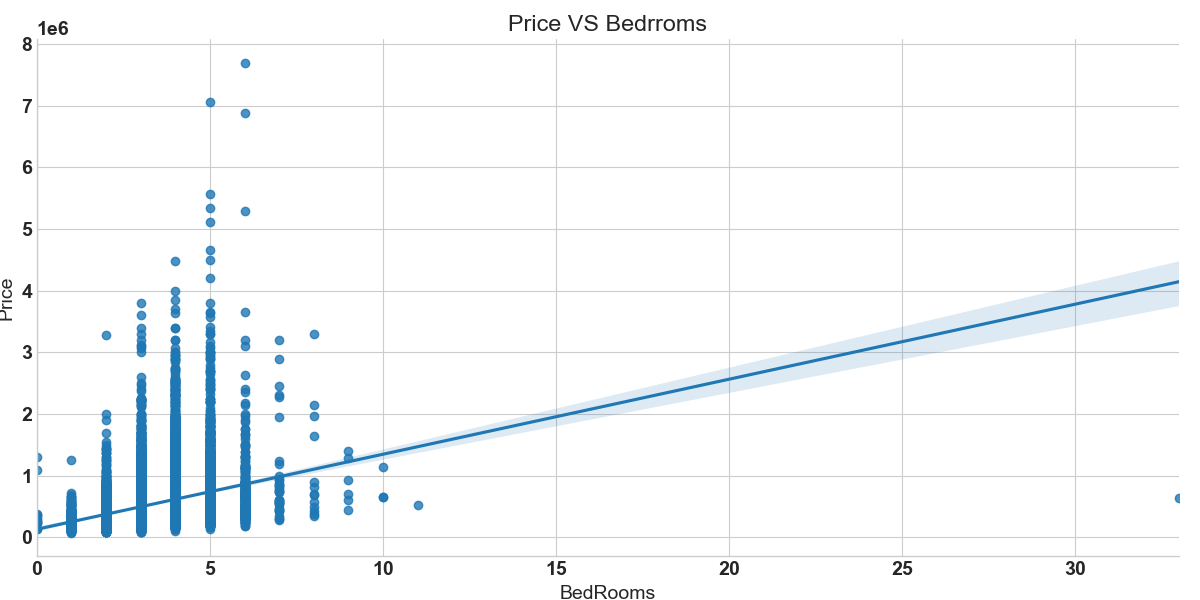


Fig 3. Linear Regression

* To implement Linear Regression, the initial stage is to preprocess the data by scaling it using Standard Scaler from the scikit-learn library. Subsequently, we fit the model to the training data and make predictions on the test data. We evaluate the model's performance using metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R2) score.

## Decision Tree Regression:

* A Decision Tree is a versatile tool that can be used for both classification and prediction tasks. The structure of a Decision Tree resembles a tree, with internal nodes representing tests on specific attributes and branches indicating the test outcomes. After creating the Decision Tree, new instances can be easily classified by following the tree structure from the root to the leaf nodes. One of the benefits of using Decision Trees for classification is that it does not require much computation. Moreover, Decision Trees can handle both continuous and categorical types of attributes.
* To apply Decision Tree Regression, the initial step is to preprocess the data by scaling it using Standard Scaler from the scikit-learn library. We then fit the model to the training data and use it to predict the housing prices in the test data. We evaluate the model's performance using metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R2) score.

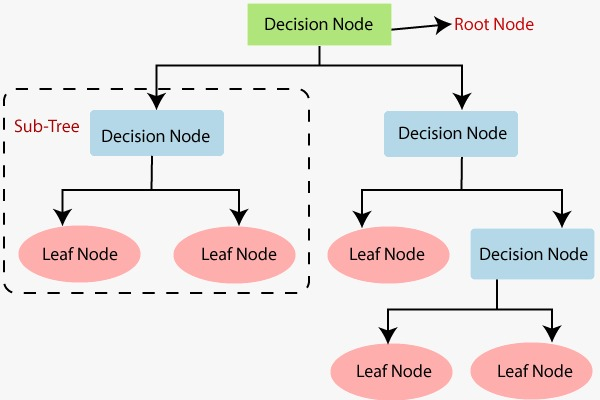


Fig 4. Decision Tree Regression

# Dataset

Regarding the King’s Country dataset, it is a well-known dataset in the field of machine learning and is often used to build predictive models for housing prices. The dataset consists of 21613 samples, each containing 21 features such as floors, bedrooms, and condition, etc. The target variable in this dataset is the price of the Houses.

To assess the effectiveness of a predictive model developed using a dataset, it is typical to partition the data into training and testing sets using a 70/30 split. The model is trained using the training data, while its performance is evaluated using the testing data. This strategy helps to prevent the model from overfitting to the training data and ensures that it can effectively generalize to new, previously unseen data.

# Experiment and Results

In this part of experiment, comparison of the accuracy levels of the Linear Regression and Decision Tree Regression models for the Housing Price Prediction scenario and show the best solution.

We construct a Scikit-learn for evaluation purposes, which allows us to compare the models using the following information:

|  |
| --- |
| regressor\_LR = LinearRegression()  regressor\_LR.fit(X\_train, y\_train)  From sklearn.metrics import mean\_squared\_error, r2\_score  y\_pred\_lin = regressor\_LR.predict(X\_test)  accuracyscore = mean\_squared\_error(y\_test, y\_pred\_lin)  R2Score=r2\_score(y\_test, y\_pred\_lin)  print("Linear Regression")  print(R2Score) |

|  |
| --- |
| regressor\_LR= DecisionTreeRegressor(random\_state=0)  regressor\_LR.fit(X\_train, y\_train)  from sklearn.metrics import mean\_squared\_error, r2\_score  y\_pred\_lin = regressor\_LR.predict(X\_test)  accuracyscore = mean\_squared\_error(y\_test, y\_pred\_lin)  R2Score = r2\_score(y\_test, y\_pred\_lin)  print("Decision Tree Regressor")  print(R2Score) |

1. EPERIMENT RESULTS

| **Sl. No.** | ***Models*** | **Rsquared(R2)** | **rmse** |
| --- | --- | --- | --- |
| 1 | **Linear Regression** |  |  |
| 2 | **Decision Tree** |  |  |

Table 1: Evaluation of Linear Regression and Decision Tree Regression

As shown in Table 1, both Linear Regression and Decision Tree Regression can be used to predict housing prices with reasonable accuracy.

Finally, we compare both Linear Regression [dbfdb] in TABLE I, including the Decision Tree Regression [dgdffb] TABLE I. The Best Model performance is sdgdg]. The best model performance is [Ridge Regression with 0.79djhgkfdjg] R-Squared value and minimum RMSE Values.

# Conclusion

The study was able to apply linear regression and decision tree regression algorithms to predict house prices using a dataset of house features. The findings showed that both algorithms were effective in predicting house prices, but decision tree regression performed better in terms of prediction accuracy. Feature selection and engineering were also found to be crucial factors in improving the models' performance. The study's contributions to the field of machine learning for real estate provide valuable insights into the suitability of various algorithms for this task. Future research should aim to expand the dataset and explore additional machine learning algorithms to enhance the accuracy and reliability of house price prediction models. Overall, this study presents a promising approach for developing more accurate and reliable models for real estate applications.

# References

1. X. Li, H. Zhang, Y. Wang, and Z. Zhang, "A Comparative Study of Ensemble Regression Models for House Price Prediction", *IEEE Access*, *2022.*.
2. J. Chougale, N. Deshmukh, A. Shinde, V. Latke and D. Sawant, “House Price Prediction using Machine learning and Image Processing”, Journal of University of Shanghai for Science and Technology, vol. 23,no. 6, June 2021.
3. Q.Truong, M. Nguyen, H. Dang and B. Mei, “Housing price prediction via improved machine learning techniques”, Procedia Computer Science, vol. 174, no. 1, pp. 433-42, 2020.
4. J. Manasa, R. Gupta and N. S. Narahari, “Machine Learning based Predicting House Prices using Regression Techniques”, 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 2020.
5. M. Ali and M. F. Hassan, “Ensemble Methods for Regression Analysis: A Case Study on House Price Prediction”, 2018 IEEE International Conference on Innovations in Information Technology(IIT).
6. M. Thamaraj and S. P. Malarvizhi, “House Price Prediction Modeling Using Machine Learning”, 2020 International Journal of Information Engineering and Electronic Business, 12(2).
7. Sdsgdsgs
8. Dsgrbrdbdrb
9. Fdbrbrdbdbd
10. Bdfbtdbtd
11. Bdtntdndfb